

Ein L^AT_EX- Skript

Martin Schimmels

18. Juli 2020

Es wird die Erlaubnis gewährt, dieses Dokument zu kopieren, zu verteilen und/oder zu modifizieren, unter den Bestimmungen der GNU Free Documentation License, Version 1.3 oder jeder späteren Version, veröffentlicht von der Free Software Foundation; mit keinen unveränderlichen Abschnitten, keinen vorderen Umschlagtexten und keinen hinteren Umschlagtexten. Eine Kopie der Lizenz ist aufgenommen in den Abschnitt mit dem Titel „GNU Free Documentation License“.

Copyright (C) 2013 Martin Schimmels

Fehler, Verbesserungen, Ergänzungen etc. an martin@mschimmels.de

- Allen, die sich für freie Software einsetzen und freie Software schreiben.
- Allen, die ihr Wissen nicht für sich behalten wollen, sondern es mit anderen teilen wollen. Wissen ist ein Gut, das sich vermehrt, wenn man es teilt.

Wenn Informationen allgemein nützlich sind, wird die Menschheit durch ihre Verbreitung reicher, ganz egal, wer sie weitergibt und wer sie erhält.*

- Allen, die verstanden haben, dass freie Software nicht ein Hirngespinnst ist, sondern wirtschaftliches und politisches Denken beeinflusst: der Gedanke, der dahinter steckt, kann Politik und Wirtschaft menschlicher machen.

- Für Kris Kristofferson:

Freedom is just another word for nothing left to lose

- Frank Mittelbach, Rainer Schöpf, Carsten Heinz, Brooks Moses, Frank Neukam, Markus Kohm, Axel Kielhorn, Yannis Haralambous, Han-Wen Nienhuys, Jan Nieuwenhuize und vielen, vielen anderen, die immer wieder neue L^AT_EX-Pakete herstellen und damit L^AT_EX lebendig halten ... und zum Glück nie fertig werden lassen.

Ich erinnere mich noch an ein Gespräch mit Kristof K., kurz bevor ich angefangen habe, L^AT_EX-Kurse an der Universität zu geben. Er konnte es nicht glauben, weil er der Überzeugung war, dass L^AT_EX ausgestorben sei ... mit all den vielen Textverarbeitungsprogrammen. Nein, Kristof, L^AT_EX lebt.

- L.D., die mal nach einem \LaTeX -Kurs gefragt hat: ohne diese Frage hätte ich vielleicht nicht angefangen zu schreiben.
- Hannah, Michael K., Valentina K. und andere freundliche Frager nach irgendwelchen Möglichkeiten von \LaTeX , die ich sonst nicht entdeckt hätte.

*Richard Stallman, Datum und Ort unbekannt

Inhaltsverzeichnis

1. Vorbemerkungen	9
1.1. Typographische Konventionen	9
1.2. Freiheit	10
1.3. Gleichberechtigung	10
1.4. Wer? Wo? Wie?	11
1.5. Und was sonst noch?	11
2. Wörter, Texte, . . .	13
2.1. Allgemeines zum Schreiben von Texten	13
2.1.1. So mach ich es jetzt!	13
2.1.2. Sammeln, sammeln, sammeln	13
2.1.3. Ordnung ins Leben bringen: die Gliederung	13
2.1.4. Der Unterschied	14
2.1.5. Was ist ein Text?	14
2.1.6. Millionen Fliegen können sich nicht irren	15
2.1.7. The Name of the Game	17
2.2. Für wen? Und was ist nicht drin?	19
3. Was braucht man?	21
3.1. L ^A T _E X-Distributionen	21
3.2. Installation	21
3.3. Editor oder IDE?	21
3.3.1. Editoren	22
3.3.2. Integrierte Entwicklungsumgebungen (IDEs)	23
4. Für die Ungeduldigen	29
4.1. Ein Standard-Rahmen	29
4.1.1. Unterschied von Textverarbeitung und Satzsystem	29
4.1.2. Vorspann und eigentlicher Text	29
4.1.3. Elementare Text-Bausteine	31
4.2. Das Erzeugen eines Dokuments	31
4.2.1. Verschiedene Arten des Aufrufs	31
4.2.2. . . manchmal auch nicht	32
4.2.3. . . und so kommt es raus	32
4.2.4. . . und L ^A T _E X produziert noch mehr	33
4.2.5. Absätze und neue Zeilen	33
4.3. Lass es so!	34

5. ... ziemlich viele Gestaltungsmöglichkeiten	35
5.1. Boxen	35
5.2. Das Nichts	36
5.3. Längenangaben	36
5.4. Zähler	38
5.5. Sonderzeichen	38
5.5.1. Kleine waagrechte Striche	38
5.5.2. Allgemeine Sonderzeichen	38
5.5.3. Zeichen mit Akzenten	40
5.5.4. Ligaturen	41
5.5.5. Anführungszeichen	41
5.6. Umgebungen und Gruppen	42
5.7. Listen	43
5.7.1. Numerierte Listen	43
5.7.2. Nicht-numerierte Listen	45
5.7.3. Beschreibungslisten	46
5.7.4. Veränderung der Listen	47
5.7.5. Mehrspaltige Listen	48
5.7.6. Allgemeine Listen	49
5.8. Tabellen	50
5.8.1. Poor-Mans Table	50
5.8.2. Richtige Tabellen	51
5.8.3. Spezialitäten	55
5.9. Die Ausgabe soll wie die Eingabe aussehen	57
5.9.1. www-Adressen	57
6. Mathematik (und ein bißchen mehr)	59
6.1. ... dafür wurde \TeX eigentlich gemacht	59
6.1.1. Mathematik-Modi	60
6.2. Standard-Mathematik	60
6.2.1. Mathematik ist einfach (zu setzen)	60
6.2.2. Höhere Mathematik	62
6.2.3. Gestapelte mathematische Ausdrücke	63
6.2.4. Gleichungen	63
6.2.5. Sätze und ähnliches	65
6.3. Erweiterte Mathematik	67
6.3.1. Pfeile	67
6.3.2. Griechische Buchstaben und andere Symbole	67
6.3.3. Zahlenmengen	69
6.3.4. Matrizen und ähnliches	69
6.3.5. Kommutative Diagramme	70
6.3.6. Stochastik	71
6.3.7. Chemie	71

7. Bilder	73
7.1. Gleitobjekte	73
7.2. Früher und heute	74
7.2.1. Das Einbinden von Bilddateien	74
7.2.2. Das Drumherum	74
7.2.3. Absolute Positionierung	76
7.3. Fließtext um ein Bild	77
7.3.1. floatingfigure	77
7.3.2. wrapfigure	79
7.4. Teilbilder	79
7.5. Spielereien	80
7.5.1. Ziertexte	80
8. Fußnoten und Randnoten	83
8.1. Fußnoten	83
8.1.1. Fußnoten in Überschriften ¹	83
8.1.2. Fußnoten im zweispaltigen Satz	84
8.1.3. Fußnoten unter Tabellen	84
8.2. Randnoten	85
8.3. Querverweise	85
8.3.1. Querverweise in pdf-Dokumenten	85
8.3.2. Querverweise auf andere Gliederungselemente	86
9. Komplexe Dokumente	87
9.1. Der erste Eindruck zählt: Die Titelseite	87
9.1.1. Die Standard-Titelseite	87
9.1.2. Die Standard-Titelseite bei Benutzung von KOMA	90
9.2. Mein Prof will ein ganz anderes Aussehen!	92
9.2.1. Eine andere Schrift	92
9.2.2. Hervorhebungen	93
9.2.3. Absätze	94
9.2.4. Abstände	95
9.2.5. Seiten	95
10. Die Gliederung eines Dokuments	97
10.1. Die Dokumentenklassen	97
10.1.1. Standardklassen	97
10.1.2. Die Klassen von KOMA-Script	97
10.2. Gliederungsstufen und Überschriften	98
10.2.1. Ein schlauer Spruch vorneweg	99
10.3. Inhaltsverzeichnis	99
10.4. Über Überschriften	100
10.5. Sonstiges	100
10.5.1. Anhang	100

¹Genau wie Fortsetzungspunkte

11. Der Kleinkram vorne und hinten dran	101
11.1. Abbildungsverzeichnis	101
11.2. Tabellenverzeichnis	101
11.3. Index (Stichwortverzeichnis)	101
11.3.1. Die Einträge	101
11.3.2. Die Erzeugung des Stichwortverzeichnisses	102
11.4. Glossar (Schlagwortverzeichnis)	103
11.4.1. Die Einträge	103
11.4.2. So wird es eingebunden	103
11.4.3. Generieren der Schlagwortverzeichnisse	104
11.5. Literaturverzeichnis	105
11.5.1. Eigene Literaturverzeichnisse	105
11.5.2. Bib $\text{T}_{\text{E}}\text{X}$	106
11.5.3. bib $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$	109
11.5.4. Besonderheiten	110
12. Was sonst noch so alles geht mit $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$	111
12.1. Präsentationen . . . die immer funktionieren	111
12.1.1. Die Klasse Beamer	112
12.1.2. Die „Titelseite“	113
12.1.3. Das Inhaltsverzeichnis	113
12.1.4. Die einzelnen Folien	114
12.2. Musik, zwei, drei	116
12.2.1. Lilypond	116
12.3. Deutsche Schriften	119
12.4. Initialen	121
12.4.1. . . zuerst in alter deutscher Schrift	121
12.4.2. Aber es geht auch bei beliebigen Schriften	121
12.5. Zeitleisten	122
12.6. Lebenslauf	122
12.7. Smileys	126
12.8. Sonderzeichen	126
A. GNU Free Documentation License	127
B. Verzeichnisse	137
Abbildungsverzeichnis	139
Listings	141
Index	147

1. Vorbemerkungen

1.1. Typographische Konventionen

- Schreibmaschinenschrift wird für Code(-Fragmente) benutzt

-

Anmerkung



Amerkungen stehen im Online-Dokument in einem blauen Kasten.

Auf eine Anmerkung wird im gedruckten Text durch den ausgestreckten Zeigefinger hingewiesen.

-

Wichtig



Wenn etwas besonders wichtig ist, steht das im Online-Dokument in einem orange-roten Kasten.

Wenn etwas besonders wichtig ist, steht das im gedruckten Text in einem Rahmen mit einem Ausrufezeichen am Rand.

-

Tip



Ein Tip steht im Online-Dokument in einem grünen Kasten.

Auf einen Tip wird im gedruckten Text ein kleines „i“ in einem Kreis hingewiesen. Tips sind auch oft Aufforderungen, hier etwas auszuprobieren, indem man ein \LaTeX -Dokument schreibt.

1. Vorbemerkungen

1.2. Freiheit

Dieser gesamte Text ist frei im Sinne der „Freien Software“. Er steht unter der „Free Documentation Licence“, die im Anhang nachzulesen ist. Das bedeutet, dass dieser Text beliebig kopiert und verbreitet werden darf. Er darf auch modifiziert werden. Einige der Rechte, so wie sie in der „Free Documentation Licence“ erläutert werden, werden hier anschließend aufgeführt.

Das Urheberrecht dieses Textes bleibt bei mir. Die wichtigste Bestimmung ist, dass dieser Text immer unter der FDL bleibt, auch wenn er von Dir verändert wird und dass die FDL immer Teil dieses Textes sein muss.

Du darfst beliebig viele Kopien dieses Textes auf beliebigen Medien machen. Auf jeder Kopie muss aber ein urheberrechtlicher Vermerk aufgeführt sein und die FDL muss Teil der Kopie sein.

Den eigentlichen Text darfst Du nach Deinen Vorstellungen bearbeiten, Teile hinzufügen, Verbesserungen anbringen, Teile weglassen, sofern Du einige Regeln einhältst:

1. Der veränderte Text muß deutlich sichtbare Vermerke enthalten, die die Veränderung, den Autor der Veränderung und das Datum der Veränderung angeben.
2. Der veränderte Text verbleibt unter der FDL und enthält die FDL.
3. Der Text darf nicht in ein proprietäres Format umgewandelt werden.
4. Eine Kopie des veränderten Textes wird mir zugesandt.

Dieser Text ist selbst in \LaTeX geschrieben.

Da ich selber es absolut unerträglich finde, wenn Fachbücher der Informatik in schlechtem Deutsch geschrieben sind,¹ bitte ich auch alle Leser, dass sie mir eventuelle sprachliche Fehler mitteilen, gerade auch wenn es nur Leichtsinnsfehler oder kleinere Ungenauigkeiten sind.²

1.3. Gleichberechtigung

In diesem ganzen Text wird immer für einen generischen Begriff nur das in der deutschen Sprache übliche grammatikalische Geschlecht benutzt. Eine Fachkraft oder sogar eine Koryphäe auf dem Gebiet des Satzes mit \LaTeX kann also durchaus auch mal ein

¹ Es ist leider betrüblich zu sehen, wie viele Autoren gerade von Fachbüchern im Bereich der Datenverarbeitung ihre Muttersprache nicht mehr beherrschen. Zeichensetzung und Rechtschreibung folgen da oft Regeln, die niemand zuvor gehört hat — und der Autor kurz danach wieder vergessen hat. Manchmal fragt man sich da, ob der Autor schon jemals ein Programm in welcher Programmiersprache auch immer geschrieben hat, wenn man weiß, mit welcher Genauigkeit man dabei sprachliche Regeln einzuhalten hat

Besonders peinlich sollte das für das Lektorat eines Verlages sein, aber auch da hat man manchmal das Gefühl, dass dort Analphabeten eingestellt worden sind. Die typische Reaktion, wenn man einen solchen Lektor auf einen Fehler hinweist, ist dann ein „Haben Sie etwa noch nie einen Fehler gemacht?“

² Ich bitte wirklich ausdrücklich darum, mir Dinge NICHT durchgehen zu lassen, die heutzutage an Schulen viel zu vielen Schülern nachgesehen werden.

Mann sein und ein Autor auch eine Frau. Vor allem bei der Bezeichnung von mehreren Personen, vulgo Plural genannt, vermeide ich so unschöne Formulierungen wie L^AT_EX-Lernende und L^AT_EX-Lernender. Sollte jemand daran Anstoß nehmen, so steht jedem Leser es frei, den Quelltext so abzuändern, dass solche generischen Begriffe in eine nach seinen Vorstellungen politisch korrekte Form umwandelt werden.³

1.4. Wer? Wo? Wie?

Dieses Buch kann man natürlich überall lesen, am Schreibtisch, vor dem Fernsehgerät am Strand Bei vielen Kapiteln ist es aber ganz sinnvoll, dass man dieses Buch am Arbeitsplatz vor sich nimmt, den Computer eingeschaltet und alles so vorbereitet, dass man sofort die eine oder andere L^AT_EX-Datei eingeben kann oder eine kleine Übung machen kann.

1.5. Und was sonst noch?

Selbstverständlich findet man im Internet ausreichend andere Anleitungen für L^AT_EX. In meinen L^AT_EX-Anfängen habe ich besonders das L^AT_EX- Kochbuch gemocht⁴.

³Wenn es sein muss, sogar mit dem grossen Binnen-I. *horribile dictu*.

⁴**latexcook**.

2. Wörter, Texte, . . .

2.1. Allgemeines zum Schreiben von Texten

2.1.1. So mach ich es jetzt!

Wie gehe ich vor, wenn ich einen Text zu schreiben habe? Etwa diesen Text über das Thema „Wie gehe ich vor, wenn ich einen Text zu schreiben habe?“

Dies ist also ein Versuch, „elektronisch“ zu schreiben. Wie immer startete ich an meinem Schreibtisch, mit viel Papier, leer oder bedruckt, Bleistift, Kugelschreiber und vielen bunten Markierstiften.

Aber dabei fiel mir auf, dass dies hier ja ein Meta-Text ist. Warum also nicht gleich den Rechner starten und die einzelnen Schritte kommentieren, während ich sie tue.

2.1.2. Sammeln, sammeln, sammeln

Erster Gedanke: wo habe ich schon mal etwas über dieses Thema gelesen? Also: Bücher raus, Zeitschriften raus, Markierstifte in die Hand, und ab geht die Post.

Aber: Später werde ich meinen Zuhörern oder Lesern sagen, woraus ich meine Weisheiten habe. Also sollte zuerst einmal eine Literaturliste angefertigt werden.

2.1.3. Ordnung ins Leben bringen: die Gliederung

Das bedeutet: hier habe ich mein erstes Element der Gliederung meines Werkes - egal ob das jetzt ein Vortrag, eine Internet-Seite oder ein gedruckter Text ist.

Also erstelle ich neben diesem Text eine zweite Datei, das Literaturverzeichnis. Übrigens: dieses Literaturverzeichnis wird gleich als $\text{T}_{\text{E}}\text{X}$ -Datei erstellt.

Das ist natürlich ein bißchen gelogen, der aufmerksame Leser oder Hörer hat es schon gemerkt: das ist nicht das erste Element der Gliederung meines Werkes, denn vorher habe ich schon gegliedert in Einleitung und dieses Kapitel hier mit der Überschrift „Sammeln, sammeln, sammeln“

Ein wichtiges Element des Schreibens wird hier besonders hervorgehoben. Ohne eine vernünftige Gliederung ist ein Sach-Text schwer zu lesen, oft unverständlich, ein literarischer Text nicht ästhetisch, nicht schön. $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ unterstützt die Gliederung eines Textes, da $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ jedem Stück Text eine Text-Art zuordnet:

- Überschrift
- Unterkapitel-Überschrift
- Aufzählung

2. Wörter, Texte, ...

- Fußnote
- Tabelle
- ... und vieles mehr

2.1.4. Der Unterschied

Den Unterschied zur Arbeit mit einer Textverarbeitung ist jetzt schon klargeworden, oder? Während eine Textverarbeitung es zulässt, dass man einfach drauf losschreibt, ohne sein Werk gedanklich zu gliedern, erwartet L^AT_EX, dass man schon weiß, was man tut, bevor man sich an den Rechner setzt. Jedes Stück Text, das man zu schreiben beabsichtigt, gehört zu „irgendwas“ : zu einer Überschrift, zu einem Absatz, zu einem Unterkapitel, zu einer Tabelle ... Das ist wie beim Deutsch-Aufsatz in der Schule: wer eine ordentliche Gliederung gemacht hat, spart Zeit beim Schreiben, und das Ergebnis ist oft eine bessere Note. Wer einfach drauflos schreibt, quält den Leser.

2.1.5. Was ist ein Text?

Die Erstellung

Ein Text ist ein fertiges Produkt, schwarz auf weißem Papier(meistens jedenfalls).

Die Arbeit des Druckers (und da zählen viele Fertigkeiten dazu, die früher von verschiedenen Personen mit verschiedener Ausbildung geleistet wurden) besteht darin, nach bestimmten Regeln auf einer weißen Fläche

- mehr oder wenig große
- rechteckige
- mehr oder wenig intensiv eingeschwärzte

Zonen zu schaffen.

Grundlage hierfür sind gewisse typografische Regeln. Also wurde der Begriff des „typografischen Grau“ eingeführt. Diese Zonen sind durch

- ihre Position auf der Seite und
- die Menge an weißer Fläche, die eine Zone von der anderen trennt,

charakterisiert. Das ergibt eine „Geographie der gedruckten Seite“. Überspitzt gesagt: das Weiße bestimmt das Aussehen des Textes, nicht das Schwarze. Dieser Formalismus beschreibt eine Seite, dann eine Menge von Seiten, ausgehend von einer bestimmten Menge von Regeln über Abstände: Abstände von Buchstaben untereinander, von Wörtern, von Zeilen, von Absätzen, Einrückungen von Absätzen usw.

Die Wirkung

Der Begriff des „typografischen Grau“ ist etwas visuelles. Der Leser erfasst das „typografische Grau“, bevor er etwas lesen muß. Aus dem Satz-Bild erkennt der Leser, welche Art von Dokument er vor sich hat. Manchmal, wenn es ein geübter Leser ist, sogar den Verlag, den Namen der Zeitung oder der Zeitschrift etc.

Das sollte man mal versuchen: verschiedene Zeitungen „typographisch grau“ einfärben und danach versuchen zu erraten, um welche Zeitung es sich handelt.

2.1.6. Millionen Fliegen können sich nicht irren

...oder wie Berndt¹ auf S. 23 schreibt: „Am Rechner werden Texte mithilfe von Textverarbeitungen erstellt. Marktbeherrschende Softwareunternehmen sorgen dafür, dass Schüler dies heutzutage bereits als eine Art Globalweisheit mit auf den Weg bekommen. Begegnungen mit Alternativen wirken dann oft auch wie Begegnungen der dritten Art.“

Warum also sollte man Texte **nicht** mit einer Textverarbeitung schreiben? Weil eine Textverarbeitung kein Satzprogramm ist. Kopka erklärt das so, dass „professionelle Verlagsdesigner ein Layout [*erstellen*], das vorrangig die leichtere Lesbarkeit und bessere Verständlichkeit beim Leser zum Ziel hat. Hinter dem Handwerk des typographischen Designs verbirgt sich die mehr als 500 jährige Erfahrung seit Gutenbergs Erfindung der Druckerkunst. Hierzu gehören die *richtigen* Größenverhältnisse und Schriftarten zwischen Kapitel-, Abschnitts-, Unterabschnittsüberschriften zum übrigen Text und deren Gliederungsnumerierung, die maximale Zeilenlänge in Abhängigkeit von der gewählten Schriftgröße, die Einrücktiefe bei Hervorhebungen durch Einrückungen, Aufzählungen und vieles mehr.“²

Han-Wen Nienhuys und Jan Nieuwenhuize³ beschreiben das auf S. 7 ihres Lilypond-Handbuches so:⁴ Das Layout sollte schön sein — nicht aus Selbstzweck, sondern um dem Leser zu helfen.

Viele „Text-Verarbeiter“ hegen den Glauben, dass sie das „schon hinkriegen“. Mit Hilfe moderner Textverarbeitungsprogramme erhält man oft auch ein Ergebnis, das auf den ersten Blick ordentlich aussieht, gut gegliedert scheint. Der Fachmann, also der Drucker oder der Layouter, erkennt aber sofort, was von einem Laien mit einer Textverarbeitung und was (auch von einem Laien) mit einem Satzprogramm geschrieben wurde.

Der mit einem Satzprogramm erstellte Text „passt einfach“ .

In der Anleitung zu KOMA-Script fand ich folgende schöne Beschreibung dessen, warum etwas falsch gemacht wird:

„Insbesondere für die Erstellung von schriftlichen Arbeiten während des Studiums findet man häufig Vorschriften, die einer typografischen Begutachtung nicht nur in

¹berndt-ldtemd2008.

²kopka-lee1991, S. 8

³lilypondUserMan.

⁴Lilypond ist eine „Untersprache“ von L^AT_EX, mit der Noten gesetzt werden

2. Wörter, Texte, ...

keiner Weise standhalten, sondern massiv gegen alle Regeln der Typografie verstoßen. Ursache für solche Regeln ist oft typografische Inkompetenz derjenigen, die sie herausgeben. Manchmal ist die Ursache auch im Ausgangspunkt begründet, nämlich der Schreibmaschine. Mit einer Schreibmaschine oder einer Textverarbeitung von 1980 ist es ohne erheblichen Aufwand kaum möglich, typografisch perfekte Ergebnisse zu erzielen. Also wurden einst Vorschriften erlassen, die leicht erfüllbar sind.⁵

Wie arbeitet man also sinnvoll?

Nach dem vorher Gesagten ist eigentlich fast klar, dass ein nicht nur „ganz ordentliches“, sondern ein richtig gutes Ergebnis nicht durch Handarbeit von Amateuren erzielt werden kann. Der Satz eines Dokuments, das dem Leser eine Freude bereitet, weil es schön aussieht **und** gut zu lesen ist, muss alle Elemente berücksichtigen, die professionelle Buchdrucker bei ihrer Arbeit anwenden. Die einwandfreie Verteilung von schwarz und weiß auf einer Seite bedeutet sehr viel Rechenarbeit — wenn man nicht einen durch jahrelanges Hinschauen geübten Blick hat (und selbst dann gelingt das nicht immer).

Eine planvolle Vorgehensweise wird von L^AT_EX vorgegeben. Nachdem man sich (inhaltlich) ein Konzept gemacht hat, schreibt man sich eine Gliederung: dieses Stück Text ist eine Überschrift, jenes eine Teilüberschrift, das nächste Stück Text ist Teil einer Tabelle, dieses gehört zu einer Aufzählung usw. Jedes dieser Stücke Text wird als zu einem Text-Typ zugehörig markiert.

Ein Satzprogramm berechnet jetzt, wieviel von den einzelnen Stücken Text in eine Zeile passen, dann, wieviele jetzt schon erzeugte Zeilen auf eine Seite passen, wie Abstände in der Horizontalen und in der Vertikalen minimal verändert werden müssen, so dass das Auge die Veränderung nicht als solche erkennt, aber das Gesamtbild der Seite doch eine Gleichmäßigkeit ausstrahlt.

Auch klar ist dabei: eine interaktive Verarbeitung ist hier sinnlos, unerwünscht, überflüssig.

⁵komascranl, S. 51

2.1.7. The Name of the Game

Zuerst kam Donald . . . ⁶

I can't go to a restaurant and
order food because I keep
looking at the fonts on the
menu.

(DONALD E. KNUTH
zitiert nach:
[http://en.wikiquote.org/
wiki/Donald_Knuth](http://en.wikiquote.org/wiki/Donald_Knuth))

Von Donald Knuth, einem amerikanischen Mathematiker, sagt man, dass er in der Frühzeit der Computer darüber entsetzt gewesen sei, wie wenig schön mathematische Texte am Computer gesetzt wurden. Als Folge davon habe er eine mathematische Auszeit genommen und sich bei Buchdruckern informiert, welche Regeln für das Setzen von Büchern sie anwenden. Dieses Regelpaket ist dann zu seinem Hauptwerk „The T_EXBook“ verarbeitet worden, in dem er das Satzprogramm T_EX konzipiert hat.

Das Wort T_EX leitet sich von der altgriechischen Vorsilbe $\tau\epsilon\chi$ ab und wird deswegen „tech“ gesprochen. Es hat zum einen die Bedeutung, die in dem Wort Technik steckt, zum anderen hat $\tau\epsilon\chi$ aber auch die Bedeutung von Schönheit und Eleganz.

Knuth hat dann sein Konzept in die „Public Domain“ gestellt, so dass jeder, der es sich zutraut, aus diesem Konzept eine T_EX-Installation bauen kann. Dabei war er von der Qualität seiner Arbeit so überzeugt, dass er demjenigen, der im ersten Jahr nach Veröffentlichung einen Fehler findet einen Dollar, dem, der im zweiten Jahr einen Fehler findet zwei Dollar, im nächsten Jahr vier Dollar usw. versprach.⁷

Inzwischen hat Knuth diese Art Kommunikation mit seinen Lesern eingestellt, nicht, weil er dadurch zum armen Mann geworden wäre — die meisten Schecks, die er ausstellen musste, wurden nämlich nie eingelöst, sondern werden von den Empfängern als ein besonders wertvolles Zeitdokument aufgehoben, gerahmt, an die Wand gehängt.

Die Arbeit mit T_EX ist allerdings nicht besonders anwenderfreundlich, da viele Befehle in T_EX sehr umständlich zu handhaben sind. Zwar hat es sich schnell herausgestellt, dass mit T_EX vor allem wissenschaftliche Arbeiten sehr sicher und in hervorragender Qualität zu erstellen sind, aber es wurde auch schnell klar, dass dabei immer wieder die selben Folgen von T_EX-Befehlen verwendet werden konnten.

. . . und dann Leslie . . .

Leslie Lamport hat genau dieses erkannt und diese immer wiederkehrenden Folgen von T_EX-Befehlen zu einem Makro-Paket geschnürt. Dieses Paket hat er L^AT_EX genannt, und diese kleine Eitelkeit, dem Paket die Anfangsbuchstaben seines Nachnamens voranzustellen, sehen wird doch mit dem größten Vergnügen: eine saubere Arbeit. Es ist so umfangreich geworden, dass es auch sehr schnell sehr erfolgreich wurde. Heutzutage

⁶Nicht Donald Duck, der größte Trottel unter den Donalds

Ergänzung 2020: Donald Duck ist nur noch der zweitgrößte Trottel

⁷Schade eigentlich, dass Microsoft dieses schöne Prinzip nicht übernommen hat.

2. Wörter, Texte, ...

schreibt kaum jemand einen Text noch in reinem $\text{T}_{\text{E}}\text{X}$, sondern fast jeder Anwender benutzt $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$.⁸

Das $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Paket ist für fast alle Belange nicht nur ausreichend, sondern sogar sehr gut geeignet. Ob Bücher, Zeitschriften-Beiträge, Briefe: alles ist komfortabel und das Ergebnis ist sehr schön.

... und inzwischen sind es ganz viele!!!

Das Konzept von $\text{T}_{\text{E}}\text{X}$ und $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, vor allem aber die Tatsache, dass beide Pakete in der Public Domain stehen, haben dazu geführt, dass es unzählige Entwickler gibt, die für noch speziellere Belange Makro-Pakete geschrieben haben. Ob für arabische Texte (von rechts nach links schreiben!!), für hebräisch, ob Notensatz oder Zeichnungen: für alle Bedürfnisse findet man im Internet eine elegante und schöne Lösung. Und sollte es die nicht geben, reicht oft schon eine Frage in einem der Foren im Internet, und irgendjemand wird antworten, dass das noch nicht fertig sei, aber dass diese Frage wieder Ansporn ist, an der Lösung des anstehenden Problems weiter zu arbeiten.

Die erste Adresse, um etwas nachzuschlagen, ist sicher das CTAN, das Comprehensive $\text{T}_{\text{E}}\text{X}$ Archive Network. Dort werden Themen diskutiert, aber auch Lösungen vorgestellt. Die deutsche Entsprechung ist DANTE⁹ die Deutschsprachige Anwendervereinigung $\text{T}_{\text{E}}\text{X}$ e.V.

In verschiedenen Sprachen bzw. in verschiedenen Kulturkreisen gibt es aber formale Besonderheiten, die eine andere Darstellung erfordern. Dazu zwei Beispiele: in Deutschland sind die Standard-Papierformate die „DIN A“ -Formate, während in den USA etwa ein Format „letter“ üblich ist (das nicht in Zentimetern und Millimetern, sondern in inches angegeben wird). Das bedeutet, dass eine deutsche Seite ein anderes Verhältnis von Länge zu Breite hat als eine amerikanische. Also muss auch schwarz und weiß anders auf einer solchen Seite verteilt werden.

Das andere Beispiel betrachtet nur Sprachen, die lateinische Buchstaben für ihre Schrift verwenden. Die relative Häufigkeit der Buchstaben unterscheidet sich von Sprache zu Sprache, so ist z. B. im Deutschen das „e“ der häufigste Buchstabe, während das „y“ selten vorkommt. Im Französischen hingegen ist das „y“ relativ häufig. Wenn eine Sprache nun viele Buchstaben mit Unterlängen (wie z.B. y, g, q, f) mit großen relativen Häufigkeiten hat, dann wird ein Zeichensatz, bei dem die Unterlängen stark ausgeprägt sind, oft nicht als schön empfunden.

Seit einiger Zeit gibt es deshalb ein Paket, das speziell für die deutsche Sprache optimiert ist. Dieses Paket firmiert unter dem Namen „KOMA-Script“ . Zum Nachschlagen sei hier KOMA-Script¹⁰ oder die Anleitung zu KOMA-Script¹¹ empfohlen.

⁸ $\text{T}_{\text{E}}\text{X}$ hat natürlich immer noch seine Bedeutung: neue Makros muss man selbstverständlich in $\text{T}_{\text{E}}\text{X}$ schreiben.

⁹siehe DANTE

¹⁰**gnther-k-kg2008**

¹¹**komascranl**

2.2. Für wen? Und was ist nicht drin?

Ich versuche, diesen Text so zu schreiben, dass der absolute Neuling einfache Dokumente in \LaTeX schreiben kann. Einfach bedeutet, dass er die elementaren Befehle von \LaTeX kennt und damit Dokumente gliedern und gestalten kann. Einige Besonderheiten werden im Laufe des Textes auch erwähnt, aber nur die, nach denen ich in den letzten Jahren gefragt wurde oder die mir selber wichtig erschienen, weil ich eine bestimmte Gestaltung eines Textes haben wollte.

Aber immer wieder wurde ich nach besonderen Gestaltungsmöglichkeiten gefragt, die ich nicht beherrschte (und zum Teil immer noch nicht kann), so dass ich oft antworten musste: „Das kann man machen, aber ich weiß nicht wie.“ Ziel dieses Textes ist es auch, den absoluten Neuling in die Lage zu versetzen, exakte Fragen zu stellen. Dann wird er sich freuen, wenn er diese Fragen stellt, dass irgendjemand im weltweiten Netz die Frage beantworten kann oder zumindest einen Ansatzpunkt dafür liefern kann, wie das anstehende Problem angegangen werden kann. So wird auch ein Anfänger bald zum \TeX niker.

3. Was braucht man?

3.1. \LaTeX -Distributionen

\LaTeX wird fast nie auf Compact Disc (CD) verbreitet, höchstens, dass mal eine CD einem Lehrbuch beigelegt wird. Unter Linux ist \LaTeX in jeder Distribution enthalten und wird oft sogar schon bei einer Standard-Installation eingespielt. Unter Windows lädt man sich die Dateien aus dem Internet und installiert sie dann.

1. Lange Zeit war auf MS-DOS-Systemen eine Installation namens „sbtex“ sehr weit verbreitet. Ob diese überhaupt noch weiter gepflegt wird? Sie hatte aber nicht den Umfang wie die ...
2. ... von dem Stuttgarter Eberhard Mattes zusammengestellte und gepflegte Distribution „emtex“. Auch diese war hauptsächlich auf MS-DOS-Rechnern zu Hause, lief aber auch auf IBM's OS/2 Betriebssystem. Diese Distribution war in den 90er-Jahren des vorigen Jahrhunderts weit verbreitet ... und ziemlich gut.
3. Auf Windows-Rechnern ist eine heute häufig anzutreffende Distribution das „MikTeX“ .
4. Unter Unixen, also auch unter Linux, ist heutzutage die „TeXlive“ -Distribution ein Standard. Unter SuSE-Linux wird diese installiert.

3.2. Installation

Die Installation ist weder unter Windows noch unter Linux ein Problem. Im Falle eines Windows-Rechners ist etwa MiKTeX eine selbstentpackende und selbstinstallierende Datei. Runterladen, entpacken, starten.

Unter den diversen Linuxen wird über die jeweilige Paket-Verwaltung TeX/LaTeX ausgewählt. Installation starten, Kaffee trinken, mit der Arbeit beginnen.

3.3. Editor oder IDE?

Man kann \LaTeX -Dokumente mit einem einfachen Editor schreiben. Und es gibt sicher noch viele Autoren, die von der Schreibmaschine her kommend einen einfachen Editor bevorzugen, weil für diese Personen die Tastatur das einzig diskutabile Eingabegerät ist, die 10-Finger-Blind-Schreiber, die glücklich sind, wenn sie die Hände nicht von der

3. Was braucht man?

Tastatur nehmen müssen.¹ Aber selbst für diesen Personenkreis sind einige technische Hilfen eines Editors nicht von der Hand zu weisen. So ist es bei \LaTeX -Dokumenten von Vorteil, wenn Schlüsselwörter farbig hervorgehoben werden („syntax highlighting“). Eine andere wichtige Arbeitserleichterung ist es, wenn Wörter, die ständig wieder vorkommen, nach der Eingabe der Anfangsbuchstaben automatisch beendet werden („auto completion“).

Was Menschen, die durch Textverarbeitungen „vorbelastet“ sind, gar nicht kennen, ist das Ein- und Ausfalten von Text („folding“): zusammengehörende Textstücke werden als Blöcke bezeichnet. Wenn man einen solchen Block beendet hat, ist es eine große Hilfe, wenn man diesen Block ausblenden kann, um mehr Übersichtlichkeit zu gewinnen.

3.3.1. Editoren

Es gibt Editoren wie Sand am Meer. Vor allem gibt es jede Menge Editoren, die zur Freien Software gehören. Fast jeder dieser Editoren hat wesentlich mehr Fähigkeiten und Eigenschaften als der Standard-Editor, der bei Windows beiliegt. Fast alle diese frei-verfügbaren Editoren beherrschen die oben genannten Dinge. Das liegt daran, dass fast jeden Programmierer irgendwann einmal der Ehrgeiz gepackt hat und er einen eigenen Editor geschrieben hat, der natürlich genau das kann, was ein Programmierer von einem Editor erwartet.

Für Editoren gilt die einfache Regel: der Editor ist der beste, mit dem ich am flüssigsten arbeiten kann. Trotzdem stelle ich hier einen Editor vor, der für verschiedene Plattformen zur Verfügung steht, konfigurierbar, klein und schnell ist.

SciTE

Ein Editor, der für Linux und Windows gleichermaßen zur Verfügung steht, ist SciTE. Er hat viele Eigenschaften, die gutes Arbeiten ermöglichen und erleichtern, wenn man schlichte Editoren vorzieht und lieber mit der Tastatur als mit der Maus arbeitet.

In SciTE wird die Syntax farbig dargestellt und diese Darstellung ist über „properties“-Dateien, die reine Textdateien sind, zu verändern. Nicht nur Schlüsselwörter, sondern auch Kommentare, Klammern und anderes wird in verschiedenen Farben angezeigt.

SciTE beherrscht das Einfalten von Blöcken, die Suche nach zueinander gehörenden öffnenden und schließenden Klammern wird unterstützt, Blöcke können auskommentiert werden und viele andere Fähigkeiten sind eingebaut. Vor allem aber ist dieser Editor nicht nur für \TeX / \LaTeX sehr geeignet, sondern auch für viele Programmiersprachen.

In der „properties“-Datei kann auch der Übersetzer und das Anzeigeprogramm für die Ausgabedatei eingestellt werden. Somit kann SciTE über diese Datei zu einer zwar schlichten, aber trotzdem leistungsfähigen IDE umgebaut werden.

Das ist SciTE in Aktion:

¹Was dann Unixer gerne erzählen: A mouse is a device used to point at the xterm you want to type in.

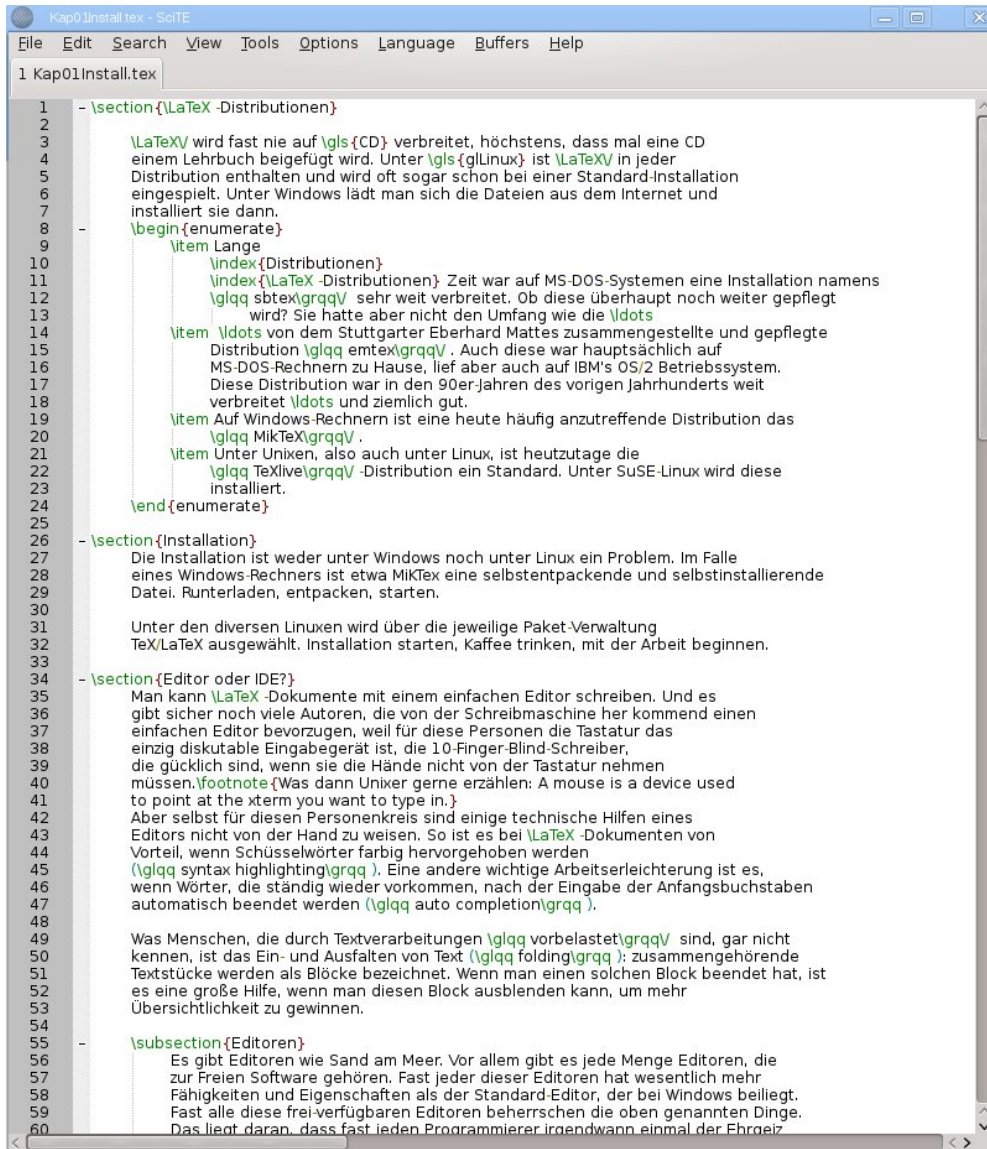


Abbildung 3.1.: Der Editor SciTE

3.3.2. Integrierte Entwicklungsumgebungen (IDEs)

Wie bei den Editoren werden hier nur IDEs vorgestellt, die frei verfügbar sind. Jede dieser IDEs hat ihre Vorteile und ihre Nachteile. Die beste IDE ist die, die dem je-

3. Was braucht man?

weiligen Benutzer bei seiner Arbeit am meisten entgegenkommt. Und wenn diese subjektiv beste IDE eine bestimmte gewünschte Eigenschaft nicht hat, gibt es mehrere Möglichkeiten:

- Da diese Programme alle Freie Software sind, kann man sich den Quelltext besorgen und die gewünschte Eigenschaft dazuprogrammieren.
- Oder man schreibt auf die Internetseite des jeweiligen Produkts seinen Wunsch, und hofft, dass andere Nutzer des Programms den selben oder einen ähnlichen Wunsch haben und dass das den Programmierer der IDE anspornt, diesen Wunsch zu erfüllen.
- Oder man probiert eine andere IDE aus.

Da aber bekanntlich der erste Eindruck wichtig ist, sind in den folgenden Abschnitten Screenshots der vorgestellten IDEs abgedruckt.

TeXMaker

TeXMaker hat den Vorteil, dass diese IDE auf allen Plattformen zur Verfügung steht. Weitere Vorteile von TeXMaker:

1. TeXMaker unterstützt Unicode. Das ist für Windows-Benutzer vielleicht nicht so wichtig, aber es ist zu hoffen, dass Unicode irgendwann zur Standard-Codierung wird.
2. TeXMaker unterstützt automatische Wort-Vervollständigung. Wer viel schreibt, wird ohne diese Eigenschaft nicht mehr arbeiten wollen.
3. TeXMaker hat vordefinierte Routinen zum Erstellen und Anschauen der Dokumente.
4. In TeXMaker integriert sind Vorlagen für verschiedene Arten von Dokumenten.
5. TeXMaker kann Teile des Textes auskommentieren und den Kommentar wieder entfernen. Das ist sehr hilfreich bei der Fehlersuche.
6. TeXMaker unterstützt den Autor bei der Bibliographie durch die Vorgabe von vielfältigen Literaturtypen.
7. TeXMaker unterstützt das Arbeiten mit komplexen Dokumenten, bei denen in eine Rahmen-Datei die einzelnen Teile des Dokuments eingebunden werden.

Weitere Informationen zu TeXMaker findet man auf der Homepage² Und so sieht TeXMaker aus:

² <http://www.xmlmath.net/texmaker>

3.3. Editor oder IDE?

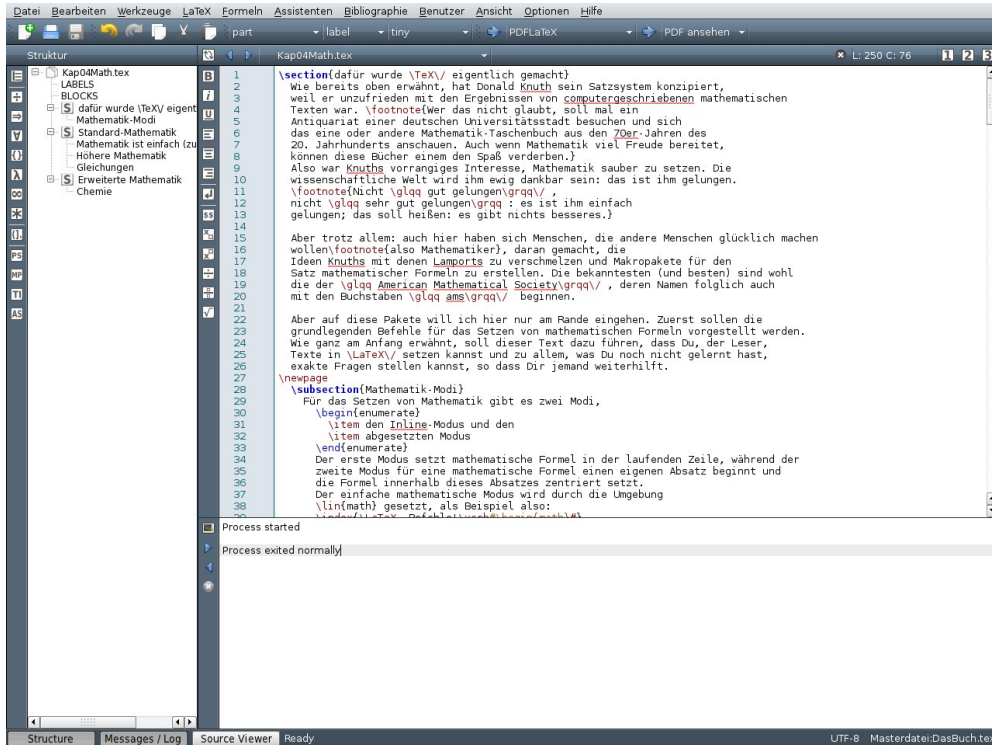


Abbildung 3.2.: Die IDE TeXMaker

TeXnicCenter

TeXnicCenter steht nur für Windows-Rechner zur Verfügung.

1. TeXnicCenter kann alle Teile von \LaTeX -Dokumenten bearbeiten.
2. Dazu müssen Profile für einzelne Bearbeitungsschritte angelegt werden.

Weitere Informationen zu TeXnicCenter findet man auf der Homepage ³

³<http://www.texniccenter.org/>

3. Was braucht man?

Kile

Kile ist als IDE für L^AT_EX unter KDE geschrieben, d.h ursprünglich unter Linux. Da aber KDE inzwischen auf alle Betriebssysteme portiert worden ist, ist es möglich, Kile auch unter Windows und MacOS zu starten.

1. Kile unterstützt Unicode. Das ist für Windows-Benutzer vielleicht nicht so wichtig, aber es ist zu hoffen, dass Unicode irgendwann zur Standard-Codierung wird.
2. Kile unterstützt automatische Wort-Vervollständigung. Wer viel schreibt, wird ohne diese Eigenschaft nicht mehr arbeiten wollen.
3. Kile hat vordefinierte Routinen zum Erstellen und Anschauen der Dokumente.
4. In Kile integriert sind Vorlagen für verschiedene Arten von Dokumenten.
5. Kile kann Teile des Textes auskommentieren und den Kommentar wieder entfernen. Das ist sehr hilfreich bei der Fehlersuche.
6. Der Editor beherrscht „folding“, d.h. dass L^AT_EX-Umgebungen ausgeblendet werden können, wenn sie nicht mehr auf dem Bildschirm zu sehen sein müssen, z.B. bestimmte Absätze.
7. Kile unterstützt die Arbeit mit Projekten. Ein Projekt ist eine Menge von zusammengehörenden Dateien, die zu einem Buch, einem Artikel etc. zusammengebunden werden. Dabei wird ein Projektverzeichnis vorgegeben, es wird eingestellt welches die Hauptdatei ist, und die Erstellung des Werkes ist weitgehend automatisiert.
8. Kile hat Menus, um die wichtigsten Sonderzeichen in den Text einzubinden. Der T_EXniker weiß zwar bald die Syntax für die wichtigsten Zeichen, aber eben doch nicht alle.
9. Kile besitzt außerdem die Darstellung der Struktur des Dokuments, besser gesagt einer Datei des Dokuments. So erhält man auf einen Blick eine Aussage, ob das Dokument ordentlich gegliedert ist oder nicht.
10. In Kile eingebaut ist eine Rechtschreibprüfung. Aber für Rechtschreibprüfungsprogramme gilt allgemein: sie sind höchstens so intelligent wie die Person, die an der Tastatur sitzt.

Wahrscheinlich ist Kile von den drei vorgestellten IDEs die mit dem größten Funktionsumfang.

3.3. Editor oder IDE?

Und so sieht Kile aus:

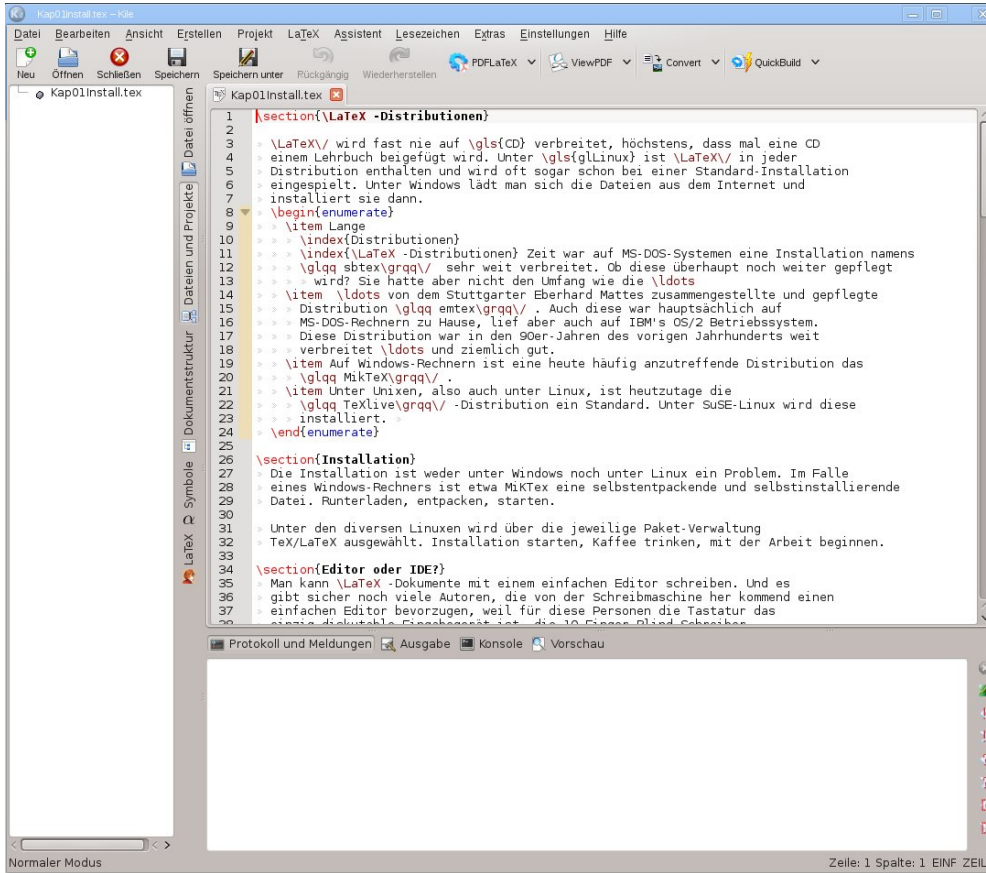


Abbildung 3.3.: Die IDE Kile

4. Für die Ungeduldigen

4.1. Ein Standard-Rahmen

4.1.1. Unterschied von Textverarbeitung und Satzsystem

Das Prinzip einer Textverarbeitung ist seit den 80er Jahren des 20. Jahrhunderts „WYSIWYG“: What You See Is What You Get.¹ Über Menus werden Textstellen mit einer Formatierung versehen, die dann diese Textstelle in einer anderen Form darstellt als ein anderes Stück Text. Dabei ist der Schreiber völlig frei in der Wahl der Mittel: Schriften, Schriftgrößen, Schriftstärken sind ohne Bindung zum Inhalt oder zur Gliederung durch den Schreiber bestimmbar. Am Bildschirm sieht man als Schreiber dann direkt, was man gemacht hat, aber das, was den Bildschirm dazu veranlasst, diese Darstellung zu realisieren (und später den Drucker, das genauso auszugeben, wie man es am Bildschirm gesehen hat) sieht man nicht.

L^AT_EX ist eine Auszeichnungssprache. Manche Leser (und Schreiber) haben mit einer anderen Auszeichnungssprache vielleicht schon zu tun gehabt: mit HTML (siehe Glossar HTML), der Sprache des Web. Das Konzept einer Auszeichnungssprache ist, dass einzelne Elemente eines Textes eine Markierung erhalten, die diese Elemente einem bestimmten Gliederungselement zuordnen. Die Markierungen sind im Text zu sehen und erst bei der Ausgabe (in eine .pdf-Datei und dann später auf den Drucker) werden diese Markierungen in Textformatierungen umgesetzt.

Dadurch werden Dinge, die zum selben Gliederungselement gehören, auch immer gleich gesetzt. Texte sehen dadurch immer gleichmäßig aus und damit schöner, und der Leser freut sich über das gleichmäßige Aussehen, weil es das Lesen erleichtert. Dieses Konzept haben Textverarbeitungsprogramme auch irgendwann übernommen. Dort heißt das dann „Formatvorlagen“.

4.1.2. Vorspann und eigentlicher Text

Egal, ob ein Editor oder eine IDE benutzt wird: jetzt muss Text eingegeben werden. Das, was wir eingeben, bezeichnen wir als „Quelltext“ . Aus diesem Quelltext macht dann L^AT_EX eine schöne Ausgabe. War bis vor einiger Zeit noch das Standard-Ausgabeformat das „dvi“ -Format (siehe auch Device Independant (DVI)), so hat sich das seit einigen Jahren geändert. Normalerweise wird heute von L^AT_EX eine Ausgabe im „pdf“ -Format erzeugt. (siehe auch Portable Document Format (PDF))

Jedes L^AT_EX-Dokument besteht aus einem Vorspann und dem eigentlichen Textteil. Der Vorspann beschreibt die Art des Dokuments, enthält Anweisungen über den Stil,

¹Und auf die Erfüllung dieses Versprechens hoffen alle, die mit Textverarbeitung ihr Werk schreiben.

4. Für die Ungeduldigen

in dem das Dokument gesetzt werden soll und bindet Pakete ein, die für das Dokument benötigt werden.

Ein einfaches Muster für einen Vorspann sieht so aus:

```
1 \documentclass[a4paper,12pt,german,ngerman,utf8]{scrartcl}
2 \usepackage[utf8x]{inputenc}
3 \usepackage{babel}
```

Listing 4.1: Minimaler Vorspann

An diesen wenigen Zeilen sind bereits allgemeine Regeln für \LaTeX zu sehen.

1. Jeder \LaTeX -Befehl beginnt mit einem Backslash „\“.
2. \LaTeX -Befehlsnamen bestehen aus Buchstaben, wobei Groß-/Kleinschreibung relevant ist. Bei vielen Befehlen gibt es sogar Versionen, die Großbuchstaben enthalten und solche, die keine enthalten.
3. Ein \LaTeX -Befehl kann Pflicht-Parameter und optionale Parameter enthalten. Pflicht-Parameter werden in geschweifte Klammern „{ }“ , optionale Parameter in eckige Klammern „[]“ geschrieben.
4. Wenn ein \LaTeX -Befehl Pflicht-Parameter enthält, müssen die geschweiften Klammern geschrieben werden — selbst wenn kein Wert mitgegeben wird. Die geschweiften Klammern bleiben bei solchen Befehlen gegebenenfalls ohne Inhalt.
5. Wenn ein \LaTeX -Befehl sich auf eine größere Menge Text auswirken soll, wird eine Umgebung geschaffen.
Eine Umgebung beginnt mit einem `\begin{Umgebungsname}` und endet mit einem `\end{Umgebungsname}`.²
6. Immer wieder stellt man fest, dass bestimmte Forderungen, die der Schreiber an den Textsatz stellt, fehlen. Von denen nimmt man aber an, dass sie bei einem so ausgefeilten System wie \LaTeX vorhanden sein sollten. Das sind sie meistens auch, aber vielleicht nicht in dem Basis-System. Wie in vielen Programmiersprachen auch, kann man ausgelagerte Teile des \LaTeX -Systems nachladen. Das geschieht durch den `\usepackage{Paketname}` -Befehl.

Die erste Zeile gibt die Dokumentenklasse an. Hier wird die in KOMA-Script für deutsche Belange angepasste Klasse für (wissenschaftliche) Artikel „scrartcl“ verwendet. Als fakultative Parameter werden

- **a4paper** für die Papiergröße DIN A 4
- **12pt** für die Standard-Schriftgröße 12 Punkt
- **german** für die Sprachanpassung an die deutsche Sprache

²zum Begriff Umgebung siehe Glossar

- **ngerman** für die Sprachanpassung an die deutsche Sprache nach der Rechtschreibreform (new german)
- **utf8** für den Zeichensatz utf8 aus Unicode

verwendet.

Der eigentliche Text steht innerhalb der „document“-Umgebung. Ein vollständiger Rahmen für ein \LaTeX -Dokument sieht also so aus:

```

1 \documentclass[a4paper,12pt,german,ngerman,utf8]{scrartcl}
2 \usepackage[utf8x]{inputenc}
3 \usepackage{babel}
4 \usepackage{graphics}
5 \begin{document}
6 \end{document}

```

Listing 4.2: Vollständiger Rahmen

4.1.3. Elementare Text-Bausteine

Wenn Du diesen Rahmen erstellt hast und \LaTeX aufrufst, dann generiert Dir \LaTeX schon ein korrektes Dokument — eine leere Seite. Also empfiehlt es sich, in das Dokument ein bißchen Text einzugeben. Schreib also irgendeinen mehr oder weniger sinnvollen Text in die document-Umgebung.³

4.2. Das Erzeugen eines Dokuments

Aus dem geschriebenen Quelltext soll jetzt ein schön aussehendes Dokument gemacht werden. Dafür ist der \LaTeX -Übersetzer zuständig. Vor langer Zeit — IT-technisch gesehen, also bis vor 20 Jahren — wurde aus dem \LaTeX -Quelltext eine sogenannte „dvi“-Datei⁴ erstellt. Diese dvi-Datei wurde dann direkt an eine Druckmaschine weitergegeben. Dieses Ausgabeformat ist inzwischen überholt: die ganze Welt benutzt „pdf“-Dateien⁵

Deswegen ist heute der Übersetzer nicht mehr das Programm „latex“, sondern das Programm „pdflatex“.

4.2.1. Verschiedene Arten des Aufrufs

Die Hardliner, Unix-Freaks, die, die nichts schöneres als eine Shell kennen, öffnen natürlich eine solche, wechseln mit ein paar „cd“’s⁶ in das Verzeichnis, in dem der Quelltext steht, und rufen dort den Übersetzer auf: `pdflatex meineDatei`

³zum Begriff Umgebung siehe Glossar Umgebung

⁴Device Independent File

⁵siehe PDF

⁶change directory

4. Für die Ungeduldigen

Die meisten Anwender eines so schönen Satzsystems benutzen wahrscheinlich aber eine ebenso schöne Entwicklungsumgebung (siehe oben auf Seite 23). Beim TeXMaker ist z.B. in der Werkzeug-Zeile als zweites von rechts ein Menüpunkt eingetragen, unter dem man die verschiedenen Übersetzer auswählen kann. Auch wenn hier standardmäßig „Schnelles Übersetzen“ angezeigt wird, rate ich davon ab. Beim schnellen Übersetzen wird das Programm „pdflatex“ aufgerufen und gleich anschließend das eingebaute Anzeigeprogramm für „pdf“-Dateien.

Bei nicht sehr umfangreichen Dokumenten funktioniert das auch nicht schlecht, aber ...

4.2.2. ... manchmal auch nicht

...man sieht die Meldungen von L^AT_EX, die leider oft genug Fehlermeldungen sind, nicht. Aber wenn man ernsthaft mit L^AT_EX arbeiten will, sollte man in der Lage sein, im Falle eines Fehlers diesem auf die Spur zu kommen. Deshalb ist es für einen Anfänger wichtig, mal einen Fehler zu machen. Dann sollte man sich in der log-Datei (wenn man auf der Kommandozeile, der Shell, arbeitet) bzw. in der Ausgabe „Meldungen/Logdatei“ der IDE auskennen und nicht verzweifeln, wenn dort Hunderte oder gar Tausende von Zeilen ausgegeben werden, die alle ganz grauenvolle Fehler vermuten lassen.

Oft ist es gar nicht so schlimm. Als Faustregel gilt beim Begutachten der Fehlermeldungen: wenn Du das Gefühl hast, dass Du gar nicht so viel falsch gemacht hast, dann betrachte nur die erste Fehlermeldung, verbessere diese und versuche Dein Glück noch einmal. Diese Vorgehensweise ist oft schon erfolgreich. Denn L^AT_EX, genauer der Übersetzer, bemerkt den ersten Fehler, und ab dieser Stelle kommt er ins Schleudern. Ein häufiger Fehler ist, dass man etwa mit `\begin{irgendeineUmgebung}` eine Umgebung begonnen hat, diese Umgebung aber nie beendet hat. Dann sucht der Übersetzer bis zum Ende des Dokuments nach dieser Ende-Markierung, und auf dem Weg zum Ende wird alles als falsch markiert, was nicht in diese Umgebung gehört.

Fast immer werden Warnungen der Form „Overful box (1.1234 pt too wide)“ generiert.⁷ Das bedeutet meistens, dass L^AT_EX beim Verteilen der Buchstaben auf die Zeile kein optimales Ergebnis hinbekommen hat. Die Angabe ist in Punkt, und ein Punkt entspricht etwa 0,37 mm. Dann kann man ja eine Faustregel aufstellen: Warnungen mit kleinen Zahlen können einfach ignoriert werden. Oder kannst Du das tatsächlich mit bloßem Auge erkennen, wenn eine Zeile 0,3 mm zu lang ist? Wenn die Zahl größer ist (und das „größer“ muss jeder einzelne für sich selbst herausfinden), sollte man sich die zum Fehler gehörende Textstelle in der Ausgabe einmal anschauen. Oft liegt einer solchen Warnung zugrunde, dass L^AT_EX nicht richtig trennen konnte. Das kann man beheben durch eine manuelle Trennhilfe (siehe dazu das nächste Kapitel).

4.2.3. ... und so kommt es raus

Was sofort auffällt: L^AT_EX fasst mehrere Leerzeichen als ein Leerzeichen auf, und ein Leerzeichen ist der Abstand zwischen zwei Wörtern. Absätze und Zeilenwechsel müssen explizit angegeben werden.

⁷Zum Thema Boxen siehe Seite 35

Dafür setzt \LaTeX ohne eine weitere Angabe immer Blocksatz, das heißt, ein beliebiges Dokument sieht aus wie dieses hier: Zeilen enden immer an der selben Stelle. Dazu verwendet \TeX zwei Hilfsmittel:

1. Der Zwischenraum zwischen Buchstaben wird für jede Zeile separat berechnet, so dass ein „geschmeidiges“ Schriftbild entsteht. Die Unterschiede in den Zeichenzwischenräumen sind dabei so minimal, dass es dem Auge nicht auffällt.
2. \TeX / \LaTeX beherrscht selbstverständlich Silbentrennung, und zwar ziemlich gut.⁸ Meistens muss man dazu gar nichts machen. Bei deutschen Texten allerdings kommt \LaTeX bei Wörtern mit Umlauten ins Straucheln: nach einer Silbe mit einem Umlaut findet \LaTeX keine Stelle mehr, an der es trennen könnte. Abhilfe schafft ein Befehl: `\-` fügt eine Trennhilfe ein. Wenn in einem mehrsilbigen Wort dieses Problem auftritt, empfiehlt es sich, nach jeder Silbe eine solche Trennhilfe einzufügen. Damit berechnet \LaTeX unter diesen Vorschlägen den optimalen.

4.2.4. ... und \LaTeX produziert noch mehr

Ein \LaTeX -Lauf generiert nicht nur die `.pdf`-Datei, die wir danach ausdrucken, verteilen ... wollen, sondern einige Dateien mehr. Manche davon schaut man (fast) nie an, andere sind hilfreich, wenn es darum geht, Fehler zu finden oder unschönes Aussehen zu verbessern. In der folgenden Tabelle sind die wichtigsten Ausgabe-Dateien zusammengefasst. Dabei gilt, dass der Dateiname der Quelldatei ohne die Erweiterung `„.tex“` mit einer neuen Erweiterung versehen wird, die auf den Inhalt hinweist.

Dateiendung	Inhalt / Bedeutung
<code>.log</code>	das Protokoll der Generierung; hier stehen auch eventuell aufgetretene Fehler und Warnungen
<code>.aux</code>	die Datei, die Querverweise auflöst
<code>.idx</code>	die Datei, die den Index (das Register am Ende eines Dokuments) enthält
<code>.toc</code>	diese Datei enthält das Inhaltsverzeichnis

Tabelle 4.1.: Dateiendungen in \LaTeX

4.2.5. Absätze und neue Zeilen

Ein Absatz ist dadurch gekennzeichnet, dass im Quelltext eine Leerzeile gelassen wird. So einfach!

Eine neue Zeile ist etwas ganz anderes als ein neuer Absatz! Während bei einem neuen Absatz normalerweise ein Abstand zum vorangehenden Absatz gelassen wird,

⁸Ich habe tatsächlich einmal ein Wort gefunden, das \LaTeX fehlerhaft trennt: das Wort „Palmolive-seife“. Da man dieses Wort in wissenschaftlichen Texten eher selten benutzt, kann dieser Fehler vernachlässigt werden.

4. Für die Ungeduldigen

hat eine neue Zeile den selben Abstand zur vorigen Zeile wie die Zeilen davor zu ihren Vorgängern. Außerdem werden Absätze meistens so gesetzt, dass die erste Zeile eines Absatzes eingezogen wird, während eine neue Zeile am linken Rand des Satzspiegels beginnt.

Eine neue Zeile wird durch den Befehl `\\` (also zwei Backslashes hintereinander) erzeugt. Das sieht dann so aus wie am Ende dieser Zeile zusammen mit dem Anfang der nächsten Zeile.

Dabei kann (aber sollte nur in seltenen Fällen) der Abstand zur nächsten Zeile als fakultativer Parameter angegeben werden. Wenn man etwa mit dem Zeilenwechsel einen vertikalen Abstand von einem Zentimeter lassen will, geschieht das durch `\\[1cm]`

4.3. Lass es so!

Eine der wichtigsten Regeln für das Schreiben mit \LaTeX ist: ändere nichts an der Darstellung. Vor allem solange der Text nicht zu Ende geschrieben ist, ändere nichts an der Darstellung. Konzentriere Dich zuerst auf den Inhalt, wenn der stimmt, konzentriere Dich auf die logische Gliederung.

Und dann lass \LaTeX den Text generieren, schau Dir das Ergebnis in Ruhe an, und überleg Dir dann, was an der Darstellung nicht passend ist. Meistens ist das Ergebnis sehr ansehnlich, denn meistens sieht es aus „wie gedruckt“, was ja auch kein Zufall ist, wenn man bedenkt, wieviel (Buchdrucker-)Intelligenz in \LaTeX steckt.

Und dann kann es so bleiben.

5. . . . ziemlich viele Gestaltungsmöglichkeiten

In diesem Kapitel werden grundlegende Befehle und Gliederungsbefehle für den Text vorgestellt. Dabei handelt es sich zuerst einmal um Textblöcke, die kein zusammenhängender Text sind. Das sind vor allem Bausteine, die etwa in einem Roman nicht benötigt werden, sehr wohl aber in wissenschaftlichen Arbeiten.

5.1. Boxen

In \LaTeX besteht alles aus Boxen¹ Trotzdem Boxen ein wesentliches Konzept des Buchsatzes sind, werden viele Leser dieses Kapitel überspringen, und das völlig zu Recht. Das Thema Boxen wird erst dann interessant, wenn Fehler in der Generierung oder im Ausdruck auftreten. Ansonsten kann man einfach darauf vertrauen: \LaTeX schreibt alles in Boxen, und das macht es so richtig, dass man sich um die Boxen nicht kümmern muss.

Die kleinste Box ist die, die ein einzelnes Zeichen enthält. Schreibt man mehrere Zeichen hintereinander, ohne ein Leerzeichen dabei zu verwenden, erhält man ein Wort². Dabei ist ein Wort nicht unbedingt etwas, was in einem Wörterbuch steht: ein Wort kann durchaus auch so aussehen: Schim12.

Diese mehreren Zeichen, die ein Wort ergeben, werden wieder in einer Box, diesmal etwas größer, untergebracht, der Wort-Box. In eine Zeile passen meistens mehrere Wörter³, weshalb also mehrere Wort-Boxen in eine Zeilen-Box gefügt werden. Spätestens hier fängt \LaTeX an zu arbeiten, das heißt zu rechnen. Denn damit eine Zeile gut aussieht, müssen ja nicht nur die Zeichen und Worte gleichmäßig und für das Leser-Auge angenehm auf die Zeile verteilt werden, sondern auch die Leerzeichen. Vereinfacht gesagt addiert \LaTeX die Länge aller Zeichen auf, und berechnet, wieviele davon plus der benötigte Zwischenraum in eine Zeile passen. Falls das nicht befriedigend gelöst werden kann, werden entweder die Zwischenräume minimal gestreckt oder geschrumpft oder es wird versucht, ein Wort am Zeilenende zu trennen, so dass Zeilenanfang und Zeilenende bündig sind.

Mehrere Zeilen-Boxen werden zu einer Absatz-Box gebündelt. Dabei muss \LaTeX noch am wenigsten rechnen, denn innerhalb eines Absatzes gibt es für ein Satzsystem den geringsten Spielraum für eine Veränderung von Abständen.

¹Spätestens hier ist klar, dass es bei „Boxen“ nicht um die Kampfsportart geht, sondern um den Plural von „Box“.

²Auch wenn Bill Gates dieses Prinzip noch nicht richtig verstanden hat.

³Gegenbeispiele findet man bei Mark Twain.

5. ...ziemlich viele Gestaltungsmöglichkeiten

Zum Schluß werden mehrere Absatz-Boxen zu einer Seiten-Box gebündelt. Auch hier werden noch zusätzliche Dinge auf eine Seite gepackt: ein Seitenkopf enthält in einem Buch oft eine Zeile mit der aktuellen Kapitelüberschrift, eine Seitenzahl findet man entweder in der Kopfzeile oder in einer Fußzeile, und hinter dem eigentlichen Text stehen oft in wissenschaftlichen Werken noch Fußnoten. So wie bei einer einzelnen Zeile in der Horizontalen Zwischenräume eingefügt werden müssen, müssen jetzt bei einer Seite in der Vertikalen Zwischenräume hinzukommen. Und wieder ist \LaTeX gefordert und rechnet aus, wie das alles schön auf einer Seite verteilt wird.

Ein typisches Beispiel für die Veränderung von Boxen ist, dass in einer Zeile etwas unterstrichen werden soll. Das sieht noch ganz gut aus.

Und das sieht ganz anders aus, gut zu sehen, wenn ich jetzt in der nächsten Zeile Mathematik mit Potenzen x^{k^2} oder viele Wörter⁴ mit Großbuchstaben oder Buchstaben mit Oberlänge benutze. Damit die Potenz noch richtig zu lesen ist und nicht die Exponenten in die darüberliegende Zeile hineinrutschen, berechnet \LaTeX hier einen etwas größeren Zeilenabstand.

Wie ich vorher schon erwähnt habe: manchmal muss der Schreiber eines Textes in das, was \LaTeX berechnet, eingreifen. Da muss manchmal eine Box in der Horizontalen oder in der Vertikalen verändert werden. Aber das sind eher Ausnahmen und diese werden hier noch nicht behandelt.

5.2. Das Nichts

Leerzeichen trennen Worte⁵. Dabei ist \TeX / \LaTeX großzügig: ob ein Leerzeichen oder mehrere ist aus der Sicht von \TeX / \LaTeX das selbe. Mehrere Leerzeichen werden wie eines interpretiert. Problematisch ist allerdings, dass \LaTeX deswegen auch denkt, dass an jedem Leerzeichen ein Zeilenwechsel stattfinden kann. Fast immer funktioniert das. In diesem Werk aber kommen Konstrukte vor wie „siehe S. 12345“. Hinter der Seitenzahl soll ein Leerzeichen stehen, aber dort soll auf keinen Fall die Zeile umbrochen werden. Das erreicht man durch das `~`, die Tilde: ein Textstück `siehe S.~12345` wird nicht zwischen der Abkürzung „S.“ und der Zahl getrennt.

Eine weitere Form des Zwischenraums ist der Zeilenwechsel (oder Zeilenvorschub) im **ingegebenen Text**. Ein solcher wird von \LaTeX wie ein Leerzeichen behandelt.

5.3. Längenangaben

Bei vielen Befehlen in \LaTeX werden Längen oder Abstände gesetzt. Meistens macht \LaTeX das richtig, aber in einigen Fällen muss der Schreiber an diesen Längen etwas ändern. Bei Aufgabenblättern für meine Schüler etwa muss manchmal alles auf eine Seite, und bei meinem ersten Versuch passt nicht alles auf eine Seite, denn es sind zu viele Aufgaben, so dass einige wenige Zeilen bewirken, dass eine weitere Seite angefan-

⁴noch dazu mit einer Fußnote

⁵Auch Namen. Kann das mal jemand Bill Gates erklären?

gen wird.⁶ Das ergibt zwar von der Seitenaufteilung her „schöne“ Seiten, aber für das Problem „Aufgabenblatt für Schüler“ ist der Aspekt „schöne“ Seiten weniger wichtig als die Vorgabe, dass die Schüler nicht auf der Rückseite eines Blattes die Fortsetzung lesen müssen. Auch wenn inzwischen viele Drucker den doppelseitigen Druck beherrschen: wer die Aufgabe gar nicht lösen will, findet den letzten Teil auf der Rückseite auch nicht.

Abhilfe schaffe ich dadurch, dass ich

- entweder jede Zeile minimal verlängere
- oder den Zeilenabstand minimal verkürze
- oder den Seitenspiegel minimal vergrößere

In allen drei Fällen ändere ich Längenangaben. Im dritten Fall ändere ich oft den oberen Seitenrand. Der dafür zuständige Befehl ist der `topmargin`. Den kann ich auf zwei Arten ändern.

- An den Befehl kann (ohne Leerzeichen, ohne ein besonderes Trennzeichen) direkt eine Längenangabe anhängen. Das sieht so aus `\topmargin-1cm`, das heißt gegenüber der Standardeinstellung verringere ich den oberen Rand um 1 cm.
- Die ausführliche Schreibweise für diese Längenänderung nutzt den \LaTeX -Befehl `setlength`. Dieser hat als ersten Parameter die zu ändernde Größe und als zweiten Parameter den neuen Wert für diese Größe. Dann sieht die oben beschriebene Änderung so aus: `\setlength{\topmargin}{-1cm}`

Den Namen der zu ändernden Größe findet man oft selbst: wie nenne ich sie auf englisch, und wie schreibe ich das eindeutig in einem Wort. Wenn man ihn nicht findet, hilft das Internet weiter.

Auch eine relative Änderung ist möglich. Dabei kann eine Länge um einen festen Betrag vergrößert oder verkleinert. Der Befehl dazu ist `\addtolength{Laengenangabe}{Zahl}`, wobei eine negative Zahl eine Verkleinerung bedeutet.

Weiterhin kann statt einer Zahl auch ein anderer Längenbefehl gesetzt werden. Dies zeige ich an einer horizontalen Linie. Zuerst kommt eine solche Linie der Länge 5 cm:

_____ Diese wird gesetzt durch `\rule{5cm}{1pt}`

Danach wird eine Linie, die halb so lang ist wie der Text einer Zeile:

_____ Das geschieht durch `\rule{.5\textwidth}{1pt}`. Beachte dabei: der Faktor „0.5“ wird ohne Leerzeichen und ohne Rechenzeichen vor die Längenangabe gesetzt.

⁶Die Schüler finden natürlich, dass es **immer viel zu viele** Aufgaben sind.

5. ...ziemlich viele Gestaltungsmöglichkeiten

5.4. Zähler

5.5. Sonderzeichen

5.5.1. Kleine waagrechte Striche

Ist Dir das auch schon begegnet? Du hörst von Leuten, von denen Du so etwas gar nicht erwartest, mitten im gesprochenen Satz auf einmal einen mathematischen Fachbegriff. Da sagt jemand „www Punkt fr Minus online Punkt de“ geschrieben: `www.fr-online.de`. Wird da etwas subtrahiert? Der \TeX niker bekommt graue Haare: ein Minus-Zeichen ist etwas anderes als ein Trennstrich, der unterscheidet sich vom Bindestrich, und der ist etwas anderes als ein Gedankenstrich. Der Trennstrich wird bei der Silbentrennung benutzt, der Bindestrich bei zusammengesetzten Wörtern wie etwa Albert–Einstein–Gymnasium und — so ein netter eingeschobener Gedanke — der Gedankenstrich dann, wenn in den fließenden Text ein Gedanke eingeschoben werden muss. Und das Minuszeichen gibt es natürlich nur in der Mathematik: $3 - 5$. Die Zeichen sind verschieden, sie sehen verschieden aus: dann sollte man auch die verschiedenen Namen für diese Zeichen benutzen. Und hier steht in einer Tabelle, wie die Zeichen gesetzt werden (auch das Minuszeichen, das eigentlich gar nicht hierher gehört, sondern in die Mathematik; aber man sieht, dass auch das anders aussieht als die anderen waagrechten Striche) und wie sie aussehen:

so heißt's	so schreibt man's	so sieht's aus	Beispiel
Trennstrich	-	-	Trenn-strich
Minuszeichen	$\$-\$$	-	$3 - 7 = -4$
Bindestrich	--	-	3 - 7 Seiten
Gedankenstrich	---	—	ein — eingeschobener — Gedanke

Tabelle 5.1.: Waagrechte Striche

Ja, ist das schön, dass verschiedene Dinge unterschiedliche Namen haben, man diese Unterschiede auch sieht⁷ und daraus unterschiedliche Bedeutungen herausliest. Hast Du übrigens bemerkt, wie schlau \LaTeX mit dem Minuszeichen umgeht? Als Rechenzeichen bekommt es einen anderen Abstand zum folgenden Zeichen als als Vorzeichen!

5.5.2. Allgemeine Sonderzeichen

Fast alle Zeichen, die auf der Tastatur zu finden sind, sind ohne weiteres in \LaTeX verwendbar. Die Ausnahmen kann man sich schon fast denken:

- der Backslash `\`

⁷wenn man ein Auge dafür hat

- alle Arten von Klammern `{}` `[]`
- und der Dollar `$`

sind für bestimmte Sachen reserviert. Deswegen können diese Zeichen nicht einfach über die Tastatur eingegeben werden. An ihrer Stelle müssen \LaTeX -Befehle verwendet werden. Die Regel dafür ist, dass entweder das Zeichen durch einen Backslash maskiert wird wie zum Beispiel für den Dollar. Der wird als `\$\$`⁸ geschrieben. Oder der Befehl wird als ausgeschriebenes Wort codiert, wie zum Beispiel für das \LaTeX -Zeichen, das als `\LaTeX` geschrieben wird.

Weitere wichtige Sonderzeichen folgen in einer Tabelle:

Schreibweise	Ergebnis	Schreibweise	Ergebnis
<code>\\$</code>	\$	<code>\S</code>	§
<code>\&</code>	&	<code>\dag</code>	†
<code>\%</code>	%	<code>\ddag</code>	‡
<code>\#</code>	#	<code>\pounds</code>	£
<code>\copyright</code>	©	<code>\AA</code>	Å
<code>\texttrademark</code>	™	<code>\oe</code>	œ
<code>\textregistered</code>	®	<code>\O</code>	Ø

Tabelle 5.2.: Sonderzeichen

⁸Der Dollar steht hier 2-mal, weil \LaTeX sonst meckert, dass der Mathematik-Modus nicht ausgeschaltet wird!

5. ...ziemlich viele Gestaltungsmöglichkeiten

Ein ganz nettes Tool im Internet findet man unter `detexify`. Hier kann man ein Symbol in eine Zeichenfläche malen, und es wird nach einem $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Symbol gesucht, das genauso (oder ähnlich) aussieht. Als Beispiel habe ich ein Bildschirmfoto von der Suche nach dem Copyright-Symbol angefügt:

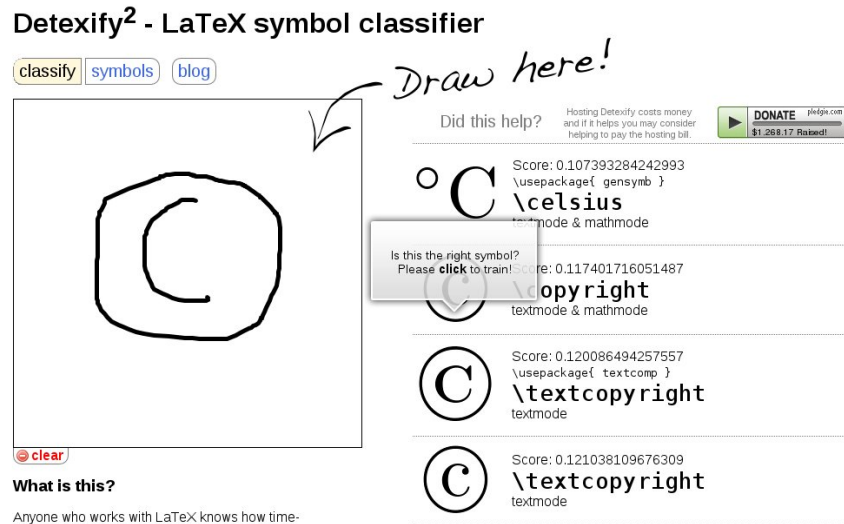


Abbildung 5.1.: `detexify`!!

5.5.3. Zeichen mit Akzenten

Auch für Zeichen mit Akzenten, wie sie in vielen romanischen aber auch in nordischen Sprachen vorkommen, gibt es eine einfache Regel (die fast immer funktioniert): das Symbol für den Akzent wird mit einem Backslash maskiert, der Buchstabe, auf den der Akzent angewendet werden soll, wird dahinter in geschweifte Klammern geschrieben. Auch das wird in einer Tabelle aufgelistet:

Schreibweise	Ergebnis	Schreibweise	Ergebnis
$\text{\' {e}}$	é	$\text{\' {e}}$	è
$\text{\^ {e}}$	ê	$\text{\c {c}}$	ç
$\text{\. {e}}$	ë	$\text{\H {e}}$	ě
$\text{\" {e}}$	ë	$\text{\u {e}}$	ě
$\text{\v {e}}$	ě	$\text{\d {e}}$	ę

Tabelle 5.3.: Akzente auf Buchstaben

5.5.4. Ligaturen

Wenn man mal „Textverarbeiter“ ärgern will, dann sollte man diesen Begriff erwähnen, vielleicht noch garniert mit der Frage, wie man denn bei manchen Wörtern Ligaturen ausschaltet. Was ist das überhaupt, eine Ligatur? Eine Ligatur ist eine Verbindung von zwei Buchstaben. Die bekannteste Ligatur in der deutschen Sprache ... kennt niemand als solche: das „ß“ ist das Zusammenziehen eines „Binnen-s“, auch Lang-S genannt, mit einem „z“ in deutschen Schriften wie zum Beispiel der „Fraktur“ (das sieht so aus: f und z gehen über in ß) oder der „Sütterlin“ (das sieht so aus: w ⁹ und r gehen über in ŕ).¹⁰ Andere Ligaturen sind das „Doppel-f“, bei dem sich die beiden kleinen runden Bögen am oberen Ende berühren, als wenn sie zusammengewachsen wären.

L^AT_EX macht es einem leicht, denn Ligaturen werden automatisch gesetzt. Schau einfach mal hier auf den „Anflug“, wie „f“ und „l“ aneinander kleben.

Das darf aber nicht immer gemacht werden. Wenn etwa „f“ und „l“ in verschiedenen Silben stehen wie in dem Wort „Auflage“, dann müssen „f“ und „l“ weiter voneinander getrennt stehen als im „Anflug“. Dies erreicht man durch die Schreibweise `Auf\lpage`. Ebenso gilt, dass bei einem von Mark Twains Lieblingswörtern, der „Schiffahrt“ die Ligatur nach dem zweiten „f“ nicht stattfinden darf. Aber so schlau ist L^AT_EX schon von Natur aus: schau einmal genau hin!

5.5.5. Anführungszeichen

In verschiedenen Sprachen werden Anführungszeichen verschieden gesetzt. Hier folgen drei Beispiele dazu und wie sie gesetzt werden:

1. „Good Evening“ entsteht durch `‘‘Good Evening’’`
2. «Bonsoir» entsteht durch `\flqq Bonsoir\frqq`
3. „Guten Abend“ entsteht durch `\glqq Guten Abend\grqq`

„lqq“ beziehungsweise „rqq“ steht für linke beziehungsweise rechte doppelte Anführungszeichen, „g“ steht für german, „f“ steht für french. Einfache Anführungszeichen werden durch ‚lq‘ gesetzt (nur ein „q“!!). Dabei ist darauf zu achten, dass hinter dem „lq“ oder „lqq“ ein Leerzeichen einzugeben ist, vor dem „rq“ oder „rqq“ aber kein Leerzeichen stehen sollte.

Außerdem gibt es bei deutschen Anführungsstrichen das Problem der Unterschneidung. Es kann passieren, dass ein Buchstabe, der auf ein schließendes Anführungszeichen folgt, teilweise unter dieses rutscht. Deswegen ist es oft angebracht, dass nach dem schließenden Anführungszeichen durch `\/` ein Zwischenraum eingefügt wird:

⁹hier musste ich ein ‚u‘ an das ‚s‘ anhängen, da ich nicht herausgekriegt habe, wie ich in Sütterlin ein alleinstehendes Lang-S ausgeben kann.

¹⁰ach, Ihr Text-Verarbeiter: wie geht so was denn mit Euren Programmen?

5. ...ziemlich viele Gestaltungsmöglichkeiten

Eingabe	Ausgabe
<code>\glqq Guten Abend\grqq\</code> sagte er	„Guten Abend“ sagte er
<code>\glqq Guten Abend\grqq sagte er</code>	„Guten Abend“sagte er

Tabelle 5.4.: Leerzeichen hinter den schließenden Anführungszeichen

Die erste Zeile sieht besser aus!

5.6. Umgebungen und Gruppen

Jedem Befehl in \LaTeX muss ein Befehlsempfänger eindeutig zugewiesen sein — und das Nichteinhalten dieser Regel ist einer der häufigsten Fehler beim Schreiben von \LaTeX -Dokumenten. Das kann auf 2 Arten geschehen: durch eine Gruppe oder durch eine Umgebung. Als Faustregel kann gelten: wenn der „Befehlsempfänger“ eine Zeichenkette von nur wenigen Zeichen ist, ist eine Gruppe angemessen, wenn es sich um einen längeren Text handelt, nimmt man eine Umgebung.

Das Wort „Umgebung“ wurde schon weiter oben benutzt, ohne dass beschrieben wurde, was eine Umgebung ist.

Eine Umgebung beginnt mit der Markierung `\begin{Umgebungsname}` und endet mit `\end{Umgebungsname}`. Dabei gilt, dass der innerhalb der Umgebung stehende Text eine andere Bedeutung hat als der außerhalb stehende und deshalb eine andere Bearbeitung verlangt. Dies wirkt sich auf das Aussehen des Textes innerhalb der Umgebung aus. Einer der häufigsten Fehler bei der Erstellung eines Dokuments mit \LaTeX ist es, dass man vergisst, eine Umgebung zu beenden ... oder dass man einen schlichten Tippfehler bei der Ende-Markierung macht.

Eine Gruppe erkennt man an den öffnenden und schließenden geschweiften Klammern. Innerhalb dieser Klammern wird eine Erklärung (engl. declaration) gegeben, die auf den Text innerhalb der geschweiften Klammern Auswirkung hat. Ein Beispiel dafür ist eine Hervorhebung eines Textabschnittes, der durch

```
{\em das wird hervorgehoben}
```

erreicht wird und im Normalfall als „italic“ dargestellt wird: *das wird hervorgehoben*.

5.7. Listen

Auf den nächsten Seiten folgen diverse Bausteine, die den regulären Textfluss unterbrechen. Das heißt, dass diese Bausteine dem betroffenen Text eine andere Bedeutung geben. Das heißt aber auch, dass dieser Text anders gesetzt werden muss.

Der erste solche Baustein sind Listen. Die treten in zwei Formen auf, als nummerierte¹¹ Listen und als nicht nummerierte Listen¹². Die zweite Art sind die, die mit Spiegelstrichen gliedern.

5.7.1. Numerierte Listen

Numerierte Listen werden durch den Befehl `\begin{enumerate}` eingeleitet und durch `\end{enumerate}` beendet. Die einzelnen Listenelemente werden durch `\item` markiert. Eine typische Liste wird also so geschrieben:

```

1  \begin{enumerate}
2  \item das ist der erste Punkt
3  \item und das der zweite
4  \item und ab hier wird es langweilig
5  \end{enumerate}

```

Listing 5.1: Eine nummerierte Liste

und sieht so aus

1. das ist der erste Punkt
2. und das der zweite
3. und ab hier wird es langweilig

So soll es sein!! Die Hausnummern der Liste verwaltet \LaTeX selbst, denn der Computer — besser gesagt dieses schlaue Programm — kann viel besser zählen als ich. Damit ist es aber auch einfach, eine solche Liste umzusortieren, ein Listenelement hinzuzufügen oder eines zu löschen. Ein Element wird genauso behandelt wie jedes andere, und das wievielte in der Liste es ist, wird von \LaTeX bei jeder neuen Generierung des Dokuments berechnet.

¹¹Der Duden führt diese Schreibweise noch, also benutze ich sie, nicht ohne mich darüber zu ärgern, dass die Kommission, die sich der „Neuen Deutschen Rechtschreibung“ angenommen hat, an manchen Stellen vielleicht doch nicht zu Ende gedacht hat. Wie man weiter unten sieht, enthält z.B. im Englischen der Wortstamm zu „Nummer“, also 'number', nur ein „m“. Folglich haben auch alle abgeleiteten Wörter nur ein „m“, wie zum Beispiel das 'enumerate'.

Auch im Deutschen besaß der Wortstamm nur ein „m“, bis die obige Kommission ... alles verbessert hat.

Wer sich die der Arbeit dieser Kommission innewohnende Intelligenz vor Augen — besser vor Ohren — führen will, soll doch mal das Teilgebiet der Mathematik „Numerik“ nach der neuen Rechtschreibung schreiben — und dann auch aussprechen. Ob die das wirklich ernst meinten?

¹²ach, welch schöne Aussage. Das freut den Logiker!

5. ...ziemlich viele Gestaltungsmöglichkeiten

Eine Bemerkung für faule Menschen (wie ich einer bin): das Einrücken von logisch von einem Strukturierungsbegriff, hier der numerierten Liste, abhängigen Konstrukten beim Schreiben des Werkes ist keine Pflicht. Es scheint auf den ersten Blick penibel, überflüssig, pedantisch, ... aber spätestens dann, wenn Fehler auftreten, ist es eine große Hilfe, dass man auf einen Blick überschaut, was zusammengehört und was von einem Gliederungsbefehl abhängt. Bei der Arbeit mit einer IDE wird man in Bezug auf das Einrücken unterstützt, gute IDE's erlauben es auch, dass man seinen ganz persönlichen Einrückungs-Stil definiert. Gute IDE's unterstützen das auch noch weiter, indem sie erlauben, dass man zusammengehörige Textstücke einklappt (folding).

Und dass \LaTeX sinnvoll einrückt ist inzwischen fast jedem klar!! Schließlich ist \LaTeX ein intelligentes Programm!

Selbstverständlich kann man solche numerierten Listen auch ineinanderschachteln: Eine numerierte Liste in einer numerierten Liste wird also so geschrieben:

```
1  \begin{enumerate}
2  \item das ist der erste Punkt
3  \begin{enumerate}
4  \item der erste Punkt hat zwei Unterpunkte ,
5      wovon dies hier der erste
6  \item und das der zweite ist
7  \end{enumerate}
8  \item und das der zweite
9  \item und ab hier wird es langweilig
10 \end{enumerate}
```

Listing 5.2: Eine geschachtelte numerierte Liste

und sieht so aus

1. das ist der erste Punkt
 - a) der erste Punkt hat zwei Unterpunkte, wovon dies hier der erste
 - b) und das der zweite ist
2. und das der zweite
3. und ab hier wird es langweilig

5.7.2. Nicht-numerierete Listen

Spiegelstrich-Listen werden durch den Befehl `\begin{itemize}` eingeleitet und durch `\end{itemize}` beendet. Die einzelnen Listenelemente werden durch `\item` markiert. Eine typische Liste wird also so geschrieben:

```

1  \begin{itemize}
2  \item das ist das erste wichtige Ereignis
3  \item und das das zweite
4  \item und noch mehr folgen
5  \end{itemize}

```

Listing 5.3: Eine Liste mit Spiegelstrichen

und sieht so aus

- das ist das erste wichtige Ereignis
- und das das zweite
- und noch mehr folgen

Selbstverständlich kann man auch solche Listen ineinanderschachteln: Eine Spiegelstrich-Liste in einer Spiegelstrich-Liste wird also so geschrieben:

```

1  \begin{itemize}
2  \item das ist der erste Punkt
3  \begin{itemize}
4  \item der erste Punkt hat zwei Unterpunkte ,
5  wovon dies hier der erste
6  \item und das der zweite ist
7  \end{itemize}
8  \item und das der zweite
9  \item und ab hier wird es langweilig
10 \end{itemize}

```

Listing 5.4: Eine verschachtelte Liste mit Spiegelstrichen

und sieht so aus

- das ist der erste Punkt
 - der erste Punkt hat zwei Unterpunkte, wovon dies hier der erste
 - und das der zweite ist
- und das der zweite
- und ab hier wird es langweilig

5. ...ziemlich viele Gestaltungsmöglichkeiten

5.7.3. Beschreibungslisten

Beschreibungslisten bestehen je Eintrag aus einem Begriff und einer Beschreibung des Begriffs. Der Begriff selber wird fett gedruckt, die Beschreibung wird normal gesetzt. Eine Beschreibungsliste wird mit der Umgebung `description` gesetzt. Die Elemente der Liste, die `items`, haben hier einen optionalen Parameter, nämlich den Begriff, der beschrieben werden soll. Als Beispiel soll hier eine Telefonliste dienen:

```
1 \begin{description}
2   \item[Karl A. Schwein] 07071 12 34 56
3   \item[Susi Sorglos] 089 66 66 66
4   \item[Donald Duck] 012 345 54 23
5 \end{description}
```

Listing 5.5: Beschreibungsliste

Das erscheint in der Ausgabe so:

Karl A. Schwein 07071 12 34 56

Susi Sorglos 089 66 66 66

Donald Duck 012 345 54 23

5.7.4. Veränderung der Listen

Vor allem bei numerierten Listen ist oft eine andere Numerierungsart gewünscht. Die Standardeinstellung ist: arabische Ziffern mit einem abschließendem Punkt für die erste Stufe, geklammerte kleine lateinische Buchstaben für die zweite Stufe, kleine römische Ziffern mit abschließendem Punkt für die dritte Stufe. Vor allem in wissenschaftlichen Texten ist aber oft eine Numerierung der Form „1.2.5.“ gewünscht.¹³

Dazu ändert man im Vorspann des Dokuments einige Zähler. Der Befehl dazu ist `\renewcommand{}{}`. Dabei steht in der ersten Klammer der zu ändernde Zähler, in der zweiten Klammer der neue Wert für diesen Zähler. Um auf der zweiten Stufe also eine Form „1.2.“ zu erreichen, muss der Zähler der zweiten Stufen geändert werden auf „Zähler-der-ersten-Stufe-Punkt-Zähler-der-zweiten-Stufe-Punkt“. Dies geschieht durch den Befehl

`\renewcommand{\labelenumii}{\theenumi.\arabic{enumii}.}` während der Befehl für die dritte Stufe so aussieht:

`\renewcommand{\labelenumiii}{\theenumi.\arabic{enumii}.\arabic{enumiii}.}`

Das soll hier demonstriert werden:

```

1  \renewcommand{\labelenumii}{\theenumi.\arabic{enumii}.}
2  \begin{enumerate}
3    \item das ist der erste Punkt
4    \begin{enumerate}
5      \item der erste Punkt hat zwei Unterpunkte ,
6        wovon dies hier der erste
7      \item und das der zweite ist
8    \end{enumerate}
9    \item und das der zweite
10   \item und ab hier wird es langweilig
11 \end{enumerate}

```

Listing 5.6: Eine geschachtelte numerierte Liste mit neuer Gestaltung

und sieht so aus

1. das ist der erste Punkt
 - 1.1. der erste Punkt hat zwei Unterpunkte, wovon dies hier der erste
 - 1.2. und das der zweite ist
2. und das der zweite
3. und ab hier wird es langweilig

¹³Der Duden sagt aber hierzu, dass hinter der letzten Ziffer bei einer solchen Schreibweise *kein* Punkt stehen darf. Also sollte man lieber noch mal den Adressaten fragen, wie die Numerierung aussehen soll.

5.7.5. Mehrspaltige Listen

Bei langen Listen (also Listen mit vielen Einträgen), bei denen die einzelnen Listenelemente nur kurz sind, bietet es sich manchmal an, die Listen mehrspaltig zu setzen. Da die Option `twocolumn` immer eine neue Seite beginnt und für eine ganze Seite gilt, ist diese Möglichkeit in dem hier geschilderten Zusammenhang nicht sinnvoll.

Aber auch hier gibt es wieder eine Lösung durch ein Zusatzpaket: das Paket `multicol`. Es enthält die Umgebung `multicols`, deren erster Parameter die Anzahl der Spalten ist. Ein Beispiel für die Benutzung dieser Umgebung findet man auf der Seite 112. Es wurde erzeugt durch

```
\begin{multicols}{3}
  \begin{itemize}
    \item AnnArbor
    \item Antibes
    \item Bergen
    etc.

  \end{itemize}
\end{multicols}
```


5.7.6. Allgemeine Listen

Die bisher vorgestellten Listen sind von L^AT_EX vorgegeben. Aber es ist natürlich möglich, seine eigene Vorstellung einer Liste zu verwirklichen. Da hierbei viele T_EX- Befehle und Größenangaben benutzt und verändert werden müssen, wird dies als „allgemeine Liste“ bezeichnet.

Hier folgt ein Beispiel, wie so etwas aussehen könnte:¹⁴

- Side A:**
1. ... And The Gods Made Love
 2. Have You Ever Been (To Electric Ladyland)
 3. Crosstown Traffic
 4. Voodoo Chile
- Side B:**
1. Little Miss Strange
 2. Long Hot Summer Night

Das wurde gesetzt durch folgende Befehle:

```
\newcounter{lpseite}
\begin{list}{\bfseries Side \Alph{lpseite}:}{\usecounter{lpseite}
\labelwidth2.5cm \leftmargin3.5cm}
\item
  \begin{enumerate}
    \item \ldots And The Gods Made Love
    \item Have You Ever Been (To Electric Ladyland)
    \item Crosstown Traffic
    \item Voodoo Chile
  \end{enumerate}
\item
  \begin{enumerate}
    \item Little Miss Strange
    \item Long Hot Summer Night
  \end{enumerate}
\end{list}
```

Hier wurde zuerst ein neuer Zähler definiert. Dann wurde die allgemeine Liste durch `\begin{list}` begonnen. Der erste Parameter beschreibt die Listenmarkierung. Hier wurde eine fette Schrift gewählt, um zum einen das Wort „Side“, zum anderen die alphabetische Numerierung mit Hilfe des oben definierten Zählers auszuwählen. Der zweite Parameter gibt die Formatierung für den Listeneintrag an. Hier wird die Breite der Markierung, die Breite des linken Randes und die Schriftart „Small Capitals“ mittels `\sc` eingestellt.

¹⁴aus dem Booklet von „The Jimi Hendrix Experience: Electric Ladyland“

5.8. Tabellen

Der nächste dieser Bausteine sind Tabellen. Die Behandlung von Tabellen ist allerdings etwas schwieriger als die von Listen. Während Listen auch noch verständlich sind, wenn sie über mehrere Seiten gehen, ist das bei Tabellen nicht selbstverständlich. Vor allem bei Messdaten (zum Beispiel in Physik, Biologie etc.) sollte bei einer Tabelle über mehrere Seiten immer auf jeder Seite eine Tabellenüberschrift stehen. Darum behandelt \LaTeX Tabellen auch anders als Listen.

Hinzukommt, dass in einem ordentlichen Dokument mit mehr als einer Tabelle oft gewünscht wird, dass ein Tabellenverzeichnis eingefügt wird. Aber da hilft \LaTeX wieder dadurch, dass der Schreiber fast nichts machen muss.

5.8.1. Poor-Mans Table

Es gibt eine einfache und damit nicht sehr komfortable Art, Tabellen zu erstellen, nämlich mit der `tabbing`-Umgebung. Im Prinzip wird hiermit nur die Tabulator-Taste einer Schreibmaschine simuliert.

Diese einfachen Tabellen schreibt man in der Art, dass man eine Musterzeile schreibt, an der sich alle folgenden Tabellenzeilen orientieren. Damit ist aber schon das Problem der `tabbing`-Vorgehensweise implizit genannt: wenn man die Musterzeile nicht gut gestaltet, wird die Tabelle grauenvoll aussehen.

Die Spalten der Musterzeile werden durch ein `\=` voneinander getrennt, die eigentlichen Datenzeilen werden durch ein `\>` gegliedert. Jede Zeile muss durch den Zeilenvorschub `\\` beendet werden. Hier folgt zuerst einmal ein Beispiel, das gut geht:

```
1 \begin{tabbing}
2 Platz \= Verein und Stadt \= Punkte \\
3 1 \> BVB Dortmund \> 66 \\
4 2 \> Bayern München \> 63 \\
5 \end{tabbing}
```

Listing 5.7: Tabelle mit `tabbing`

ergibt

```
Platz Verein und Stadt Punkte
1   BVB Dortmund 69
2   Bayern München 63
```

Und das sieht richtig gut aus; satztechnisch. ¹⁵

¹⁵und sportlich auch, sagen die BVB-Fans; Stand vom 12.4.2012

Jetzt wird die Tabelle „zerstört“.

```

1 \begin{tabbing}
2 Platz \= Verein \= Punkte \\
3 1 \> BVB Dortmund \> 66 \\
4 2 \> Bayern München \> 63 \\
5 \end{tabbing}

```

Listing 5.8: tabbing mit fehlerhafter Musterzeile

ergibt

```

Platz Verein Punkte
1    BVB Dortmund
2    Bayern München

```

Das Wort „Verein“ ist kürzer als die 2 Wörter „Bayern München“. Aus diesem Grund wird die Punktzahl der Vereine in der Zeile dorthin geschrieben, wo das Wort „Verein“ aufhört, also mitten in den Vereinsnamen hinein. Damit ist die Information nicht mehr lesbar¹⁶.

5.8.2. Richtige Tabellen

Eine Bemerkung muss hier vorneweg gemacht werden: in Zeiten, in denen fast jeder schon mal eine Tabelle gesehen hat, die mit einem Tabellenkalkulationsprogramm geschrieben wurde, ist das Auge mit den ersten Tabellen, die hier unten folgen werden, nicht einverstanden. Man ist es inzwischen gewöhnt, dass Tabellen-Zellen durch waagrechte und senkrechte Linien voneinander getrennt sind.

Im Buchsatz ist das nicht üblich¹⁷, eine klassische¹⁸ Tabelle verzichtet auf diese Linien, wenn es nicht unbedingt nötig ist und setzt eine Tabelle so, dass das Auge sich an den Spalten festhalten kann, ohne dass Hilfslinien gezogen werden.

¹⁶und verärgert die Fußballfans, besonders die des BVB

¹⁷um nicht zu sagen verpönt

¹⁸elegante!!

5. ...ziemlich viele Gestaltungsmöglichkeiten

Wenn man Tabellen setzen will, die komplizierte Formate haben, muss man (wenigstens) die „tabular“-Umgebung benutzen. Eine Tabelle mit drei linksbündigen Spalten (und zwei Zeilen) sieht dann so aus:

```
1 \begin{tabular}{lll}
2   a & b & c \\
3   x & y & z \\
4 \end{tabular}
```

Listing 5.9: Tabelle mit 3 Spalten

Das Ergebnis ist

```
a b c
x y z
```

Die Regeln für Tabellen sind hier zu erkennen: nach dem Beginn der tabular-Umgebung wird in einer geschweiften Klammer ein Muster für die Spalten angegeben. Es gibt die Standard-Muster

- „r“ für rechtsbündig
- „l“ für linksbündig
- „c“ für zentriert

Die einzelnen Spalten werden durch ein Ampersand (&) voneinander getrennt, die Zeilen durch einen Zeilenvorschub beendet.

L^AT_EX berechnet in diesem Fall die Breite der jeweiligen Spalten dadurch, dass jede Spalte so breit gemacht wird, dass der größte Inhalt reinpasst.

Es passiert trotz alledem, dass Tabellenspalten zu breit sind und dadurch die Tabelle entweder von der Breite her gar nicht mehr auf die Seite passt oder unschön aussieht. Für diesen Fall gibt es eine weitere Format-Anweisung, das „p“. „p“ gibt eine absolute Breite einer Tabellenspalte vor. Diese wird in einer geschweiften Klammer angegeben. Will man also eine Tabelle mit drei je 5 cm breiten Spalten setzen, so geschieht das durch

```
1 \begin{tabular}{p{5cm}p{5cm}p{5cm}}
2   a & b & c \\
3   x & Hier steht ein ewig langer Text, der in eine normale
4       Zelle einer Tabelle gar nicht passt; deswegen wird
5       die Tabelle mit festen Spaltenbreiten gesetzt! & z \\
6 \end{tabular}
```

Listing 5.10: Tabelle mit fester Spaltenbreite

und sieht so aus:

a	b	c
x	Hier steht ein ewig langer Text, der in eine normale Zelle einer Tabelle gar nicht passt; deswegen wird die Tabelle mit festen Spaltenbreiten gesetzt!	z

Waagrechte Trennlinien werden durch den Befehl `\hline` erstellt. Die obige Tabelle mit waagrechten Strichen nach jeder Zeile sieht also so aus:

```

1  \begin{tabular}{p{4cm}p{4cm}p{4cm}}
2    a & b & c \\ \hline
3    x & Hier steht ein ewig langer Text, der in eine normale
4      Zelle einer Tabelle gar nicht passt; deswegen wird
5      die Tabelle mit festen Spaltenbreiten gesetzt!
6      & z \\ \hline
7  \end{tabular}

```

Listing 5.11: Tabelle mit waagrechten Strichen

Das Ergebnis ist

a	b	c
x	Hier steht ein ewig langer Text, der in eine normale Zelle einer Tabelle gar nicht passt; deswegen wird die Tabelle mit festen Spaltenbreiten gesetzt!	z

Die oben stehende Formatierungsbeschreibung ist eine dreifache Wiederholung der selben Formatierungszeichenkette. Jeder Mathematiker oder Informatiker wird sofort nach einer Schreibweise suchen, die mit weniger Wiederholungen verbunden ist. Die Schreibweise `{p{4cm}p{4cm}p{4cm}}` kann ersetzt werden durch `{*{3}{p{4cm}}}`. Der Stern gibt an, dass eine Wiederholung folgt. Der erste Parameter ist die Anzahl der Wiederholungen, der zweite Parameter die Formatierungsanweisung für jede der Wiederholungen.

Regeln für den Satz von Tabellen sagen aus: Senkrechte Striche haben in Tabellen nichts verloren! Trotzdem ist es manchmal erforderlich oder wünschenswert, senkrechte Linien als Zellenbegrenzungen einzufügen. Dies geschieht in der Formatierungsbeschreibung durch `{|r|r|r|}`. Als Beispiel wird hier doch die Tabelle aus dem letzten Beispiel mit senkrechten und waagrechten Strichen gesetzt, also so:

5. ...ziemlich viele Gestaltungsmöglichkeiten

```
1 \begin{tabular}{|p{4cm}|p{4cm}|p{4cm}|}  
2   a & b & c \\ \hline  
3   x & Hier steht ein ewig langer Text, der in eine normale  
4     Zelle einer Tabelle gar nicht passt; deswegen wird  
5     die Tabelle mit festen Spaltenbreiten gesetzt!  
6     & z \\ \hline  
7 \end{tabular}
```

Listing 5.12: Tabelle mit senkrechten Strichen

Die Spaltenformatierung kann natürlich nach dem oben gesagten abgekürzt werden zu `{*3}{|p{4cm}|}` (beachte die beiden senkrechten Striche: der erste gehört zu jeder der 3 Spalten und steht links von jeder; der zweite ist der, der noch am rechten Rand fehlt!). Das Ergebnis ist

a	b	c
x	Hier steht ein ewig langer Text, der in eine normale Zelle einer Tabelle gar nicht passt; deswegen wird die Tabelle mit festen Spaltenbreiten gesetzt!	z

5.8.3. Spezialitäten

Hier kommen jetzt einige weitergehende Möglichkeiten des Satzes von Tabellen. Im ersten Teil werden in L^AT_EX eingebaute Fähigkeiten benutzt. Später dann werden zusätzliche Pakete für die Gestaltung von Tabellen vorgestellt.

Linien

Waagrechte Linien werden durchaus manchmal gezogen, unter anderem, um eine Überschriftszeile von den Datenzeilen abzugrenzen. Horizontale Linien werden durch `\hline` gezogen.

Manchmal ist es wichtig, eine horizontale Linie über eine Teilmenge aller Spalten zu ziehen. Dafür steht der Befehl `\cline{m-n}` zur Verfügung, wobei `m-n` angibt, dass die Linie vom linken Rand der `m`-ten Spalte bis zum rechten Rand der `n`-ten Spalte gehen soll.

```

1 \begin{tabular}{*{4}{|l|}}
2   & Mo. & Di. & Mi \\
3   7:00 & Waldlauf & & \\
4   8:00 & & Lin. Alg. I & \\
5   9:00 & & & \\
6 \end{tabular}

```

Listing 5.13: Horizontale Linien

ergibt die folgende Tabelle:

	Mo.	Di.	Mi
7:00	Waldlauf		
8:00		Lin. Alg. I	
9:00			

Das ist zuerst einmal gar nicht schlecht. Aber beim zweiten Hinschauen bemerkt man, dass die Einträge in den Zellen mit waagrechten Begrenzungslinien zu nah an diese Linien heranreichen. Also ändere ich das:

	Mo.	Di.	Mi
7:00	Waldlauf		
8:00		Lin. Alg. I	
9:00			

5. ... ziemlich viele Gestaltungsmöglichkeiten

Das sieht schon ein bißchen besser aus. Erreicht wurde es durch

```
1 \begin{tabular}{*{4}{|l|}}
2   & Mo. & Di. & Mi \\ \hline \hline
3   7:00 & Waldlauf & & \rule[-.2ex]{0mm}{3ex} \\ \cline{2-3}
4   8:00 & & Lin. Alg. I & \rule[-.2ex]{0mm}{3ex} \\
5   9:00 & & & \rule[-.2ex]{0mm}{3ex} \\
6 \end{tabular}
```

Listing 5.14: Stützen

Der Befehl, der das realisiert, sieht ganz schön knifflig aus! `\rule` baut eine senkrechte Linie, eine sogenannte Stütze. Der erste Parameter, hier das `-.2ex` gibt an, wie weit ober- oder unterhalb der normalen Grundlinie die Stütze aufsitzen soll. Die negative Angabe `-.2ex` sagt aus, dass das bei 20 % der Höhe eines normalen „x“ sein soll. Der zweite Parameter, die `0mm`, geben die Breite der Stütze an. Da man diese Stütze nicht sehen soll, wird sie in 0 mm Breite gewählt. Der dritte Parameter, die `3ex`, besagt, dass die Stütze insgesamt so hoch sein soll wie 3 normale „x“.

Es reicht dabei aus, dass man eine solche Stütze einmal pro Zeile anbringt. Diese Stütze ist stark genug, die ganze Zeile anzuheben.

Einträge über mehrere Spalten

Oft müssen in einer Tabelle – oder in einer Tabellenüberschrift – Einträge über mehrere Spalten gemacht werden. Dies geschieht durch `\multicolumn{2}{c}{Titel}`. Der erste Parameter gibt an, über wieviele der ursprünglichen Spalten sich der Eintrag erstrecken soll. Der zweite Parameter ist eine Formatierungsanweisung, der dritte ist der Text, der gesetzt werden soll. Ein Beispiel hierzu:

```
1 \begin{tabular}{llll}
2   Verein & \multicolumn{3}{c}{Spiele} & \\
3   & gesamt & gewonnen & verloren \\ \hline
4   VfB Friedrichshafen & 12 & 11 & 1 \\
5   Unterhaching & 12 & 10 & 2 \\
6 \end{tabular}
```

Listing 5.15: Einträge über mehrere Spalten

ergibt

Verein	Spiele		
	gesamt	gewonnen	verloren
VfB Friedrichshafen	12	11	1
Unterhaching	12	10	2

5.9. Die Ausgabe soll wie die Eingabe aussehen

Es gibt immer wieder Aufgabenstellungen, bei denen der eingegebene Text nicht verändert werden soll. Im vorigen Kapitel etwa sollen die \LaTeX -Befehle, die in den umrahmten Kästen stehen, nicht als \LaTeX -Befehle interpretiert werden, denn sie sollen ja gerade Dir zeigen, wie man etwas in \LaTeX schreibt. Dazu gibt es wieder verschiedene Möglichkeiten.

Die im Standard vorgegebene und sehr häufig benutzte Art, einen Text unverändert zu lassen, ist die `verbatim`-Umgebung. Sie wird also durch `\begin{verbatim}` begonnen und durch `\end{verbatim}` beendet. Alles, was innerhalb dieser Umgebung steht, wird unbesehen übernommen: Leerzeichen, Zeilenvorschübe, Absätze, einfach alles.

Für ein kürzeres Stück Text, in diesem Aufschrieb zum Beispiel ein einzelner \LaTeX -Befehl, der innerhalb einer Zeile hervorgehoben werden soll, gibt es den \LaTeX -Befehl `\verb#`. Der Lattenzaun (`#`) kann durch ein beliebiges anderes Zeichen ersetzt werden. Er ist das Signal für den Anfang des Stückes Text und er muss das Stück Text auch beenden. Damit ist auch klar: der Lattenzaun bzw. das andere Zeichen, das benutzt wird, darf im Text nicht vorkommen!

Ich will hier noch zwei Pakete erwähnen, die die Funktionalität dieser Umgebung bzw. dieses Befehls erweitern. Das eine ist das Paket `fancyvrb`, das andere das Paket `listings`. `fancyvrb` stellt die Umgebung `Verbatim` (mit großem „V“) zur Verfügung, die es ermöglicht, optionale Parameter anzufügen. Damit können Schriftgröße, Schriftart und Numerierung des wörtlichen Textes verändert werden. Außerdem kann nach Einbinden dieses Pakets auch eine wörtliche Wiedergabe innerhalb von Fußnoten erreicht werden.

`listings` habe ich benutzt, um die \LaTeX -Code-Stücke in diesem Aufschrieb zu setzen. Es bietet viele Formatierungsmöglichkeiten für das Aussehen des wörtlich wiederzugebenden Textes. Zuerst einmal gibt es den Befehl `\lstinline` für ein Stück Text in der laufenden Zeile und die Umgebung `\lstlisting`, mit der man mehrere Zeilen wörtlich zu übernehmenden Textes setzt. `listings` ist speziell für die Aufnahme von Programm-Code, den „listings“, gedacht, und deswegen stellt `listings` als Parameter Spracheinstellungen für viele Programmiersprachen zur Verfügung. Sehr zu empfehlen ist die Lektüre der Anleitung von `listings`.

Bei all diesen Möglichkeiten, einen Text unverändert zu lassen, muss aber beachtet werden: weder die Umgebungen noch die Befehle dürfen in einem anderen \LaTeX -Befehl enthalten sein.

5.9.1. www-Adressen

... setzt man mit `\url`, damit man keine Klimmzüge mit Unterstrichen, Schrägstrichen u.ä. machen muss.

6. Mathematik (und ein bißchen mehr)

6.1. ... dafür wurde T_EX eigentlich gemacht

A mathematical formula should
never be “owned” by anybody!
Mathematics belong to God.¹

(DONALD E. KNUTH)

Wie bereits oben erwähnt, hat Donald Knuth sein Satzsystem konzipiert, weil er unzufrieden mit den Ergebnissen von computergeschriebenen mathematischen Texten war.² Also war Knuths vorrangiges Interesse, Mathematik sauber zu setzen. Die wissenschaftliche Welt wird ihm ewig dankbar sein: das ist ihm gelungen.³

Aber trotz allem: auch hier haben sich Menschen, die andere Menschen glücklich machen wollen⁴, daran gemacht, die Ideen Knuths mit denen Lamports zu verschmelzen und Makropakete für den Satz mathematischer Formeln zu erstellen. Die bekanntesten (und besten) sind wohl die der „American Mathematical Society“, deren Namen folglich auch mit den Buchstaben „ams“ beginnen.

Aber auf diese Pakete will ich hier nur am Rande eingehen. Zuerst sollen die grundlegenden Befehle für das Setzen von mathematischen Formeln vorgestellt werden. Wie ganz am Anfang erwähnt, soll dieser Text dazu führen, dass Du, der Leser, Texte in L^AT_EX setzen kannst und zu allem, was Du noch nicht gelernt hast, exakte Fragen stellen kannst, so dass Dir jemand weiterhilft.

¹zitiert nach: http://en.wikiquote.org/wiki/Donald_Knuth

²Wer das nicht glaubt, soll mal ein Antiquariat einer deutschen Universitätsstadt besuchen und sich das eine oder andere Mathematik-Taschenbuch aus den 70er-Jahren des 20. Jahrhunderts anschauen. Auch wenn Mathematik viel Freude bereitet, können diese Bücher einem den Spaß verderben.

³Nicht „gut gelungen“, nicht „sehr gut gelungen“: es ist ihm einfach gelungen; das soll heißen: es gibt nichts besseres.

⁴also Mathematiker

6. Mathematik (und ein bißchen mehr)

6.1.1. Mathematik-Modi

Für das Setzen von Mathematik gibt es zwei Modi,

1. den Inline-Modus und den
2. abgesetzten Modus

Der erste Modus setzt mathematische Formel in der laufenden Zeile, während der zweite Modus für eine mathematische Formel einen eigenen Absatz beginnt und die Formel innerhalb dieses Absatzes zentriert setzt. Der einfache mathematische Modus wird durch die Umgebung `math` gesetzt, als Beispiel also:

```
1 \begin{math}
2   x = a + b
3 \end{math}
```

Listing 6.1: Einfacher Mathematik-Modus

Die Kurzformen für den einfachen mathematischen Modus sind

- das Dollarzeichen oder
- die Klammerung durch `\(... \)`

die für den abgesetzten mathematischen Modus

- zwei aufeinanderfolgende Dollarzeichen oder
- die Klammerung durch `\[... \]`

Beachte! Die zwei Dollarzeichen sind verpönt!!

6.2. Standard-Mathematik

6.2.1. Mathematik ist einfach (zu setzen)

Die Hauptregel für den Satz von Mathematik ist: schreib es so, wie Du es als (englisch-sprechender) Mathematiker sagen würdest. Die üblichen arithmetischen Zeichen „+“ für die Addition, „-“ für die Subtraktion werden einfach über die Tastatur eingegeben. Der Malpunkt für die Multiplikation wird nicht ausdrücklich geschrieben. Falls er doch gesetzt werden muss, steht der Befehl `\cdot` (centered dot, ein zentrierter Punkt) zur Verfügung. Und Divisionen werden natürlich als Bruch geschrieben.

Der Befehl dafür ist `\frac{Zähler}{Nenner}`. Weil das so schön aussieht, sollte das jetzt gleich einmal geübt werden. Die Eingabe des Bruches $\frac{3}{4}$ wird also durch `\frac{3}{4}` geschrieben.

Es spricht natürlich nichts gegen Doppelbrüche (oder noch mehr). Was sollte sich schon groß ändern: dann ist der Zähler halt auch einfach ein Bruch und der Nenner ebenso. $\frac{\frac{\text{Z. d. Z.}}{\text{N. d. Z.}}}{\frac{\text{Z. d. N.}}{\text{N. d. N.}}}$ ⁵

Auch dazu ein kleines Beispiel:

$$\frac{\frac{a}{x+y} - \frac{b}{x-y}}{2}$$

schreibt man als

$\frac{\frac{a}{x+y} - \frac{b}{x-y}}{2}$.

Um Brüche zu kürzen benötigt man das Paket „cancel“. Ein Beispiel für das Kürzen von Brüchen sieht so aus:

$$\dots = 9a^7 - \frac{2\cancel{a^2} \cdot (a^5 - 3)}{\cancel{a^2}^1} + 3a^2$$

Das wurde gesetzt durch

$\dots = 9a^7 - \frac{2\cancel{a^2}\{a^4\}\cdot (a^5 - 3)}{\cancel{a^2}^1} + 3a^2$

Ähnlich gesetzt wie Brüche werden Binomialkoeffizienten. Die Befehle dazu sind allerdings nicht in Standard- \LaTeX enthalten, sondern gehören zu den Ergänzungen der „American Mathematical Society“, die mit `\usepackage{amsmath}` zugeladen werden müssen. Dann schreibt man den Binomialkoeffizienten „6 über 3“ als `\binom{6}{3}`, und das sieht so aus: $\binom{6}{3}$. In einer etwas größeren Schrift geschieht das durch den Befehl

`\dbinom{6}{3}`: $\binom{6}{3}$

Kommen wir also zu etwas weniger trivialer Mathematik. Jeder kennt die allgemeine Lösungsformel für quadratische Gleichungen⁶. Kein Problem! Man muss wissen, dass die Quadratwurzel auf englisch die squareroot ist, der dazugehörige \LaTeX -Befehl folglich `\sqrt{}` und das Plus-Minus-Zeichen, auch ganz einfach zu raten, `\pm`.

Das Tieferstellen wird durch einen Unterstrich eingeleitet, das Hochstellen durch einen accent circonflexe. Dabei muss man beachten, dass für den Fall, dass nicht nur ein einzelnes Zeichen hoch- oder tiefgestellt werden soll, die Zeichenfolge in geschweifte Klammern gesetzt werden muss. Also sieht die Formel so aus

$$x_{1/2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

und wird so geschrieben $x_{1/2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$.

Okay, das sieht auf den ersten Blick recht wild aus. Aber wenn man sich das vorliest, ist es klar; und wenn man die Regeln, die ja wirklich sehr anschaulich sind, gelernt hat, schreibt man so eine Zeile schneller, als dass man sie sich in einer Textverarbeitung zusammengedrückt hat.

⁵Z.d.Z.: Zähler des Zählers
N.d.Z.: Nenner des Zählers
usw.

⁶In einem Bundesland, in dem man alles kann außer Hochdeutsch, auch Mitternachtsformel genannt.

6.2.2. Höhere Mathematik

Die Mathematik ab der Oberstufe des Gymnasiums bekommt ein eigenes Kapitel, nicht nur weil es eine andere Mathematik als die der vorhergehenden Klassen ist,⁷ sondern auch weil es durch die Infinitesimalrechnung zu ganz anderen Notationen gekommen ist.

Wie den meisten schon aufgefallen ist: im Mathematik-Modus werden andere Zeichensätze benutzt als im normalen Textmodus.⁸

Also setze ich die Zeile „*x* berechnet sich aus *a* und *b* durch $x = a * b + a$ “, im Mathematik-Modus. Dann siehst Du, was hier typographisch falsch laufen kann! Das kriegt man dadurch in den Griff, dass man im Mathematik-Modus die Texte, also das, was nicht Mathematik ist, in eine `mbox` schreibt. Die obige Zeile wird also gesetzt durch `$x \mbox{ berechnet sich aus a und b durch } x = a * b + a$` und sieht dann korrekt aus:⁹

$$x \text{ berechnet sich aus } a \text{ und } b \text{ durch } x = a * b + a$$

Auch Funktionsnamen dürfen natürlich nicht im Mathematik-Zeichensatz gesetzt werden. Probiere mal aus, wie schrecklich es aussieht, wenn man im Mathematik-Modus die Sinus-Funktion einfach als `sin` schreibt: $\sin(x) = 0,5$

Aus diesem Grund gibt es im Mathematik-Modus die Funktionsnamen `\sin`, `\cos`, `\ln`, `\tan` und viele mehr. Eine einfache Methode, das nachzuschauen, ist der Menüpunkt Math → Mathematische Funktionen in der IDE Texmaker.

Eine wichtige Gruppe von Formeln in der Analysis sind die, die Integrale enthalten. Das Integralzeichen selbst ist wieder recht einfach: `\int`. Allerdings ergeben sich als Folge zwei Möglichkeiten, ein Integral zu setzen.

1. `\int_0^1` setzt die obere und untere Grenze seitlich an das Integralzeichen.
2. `\int\limits_0^1` setzt die obere und untere Grenze über und unter das Integralzeichen.

Was hier über das Integralzeichen gesagt wurde, gilt entsprechend auch für das Summenzeichen und das Produktzeichen.

Die Unterschiede sind hier anschließend zu sehen:

$$\int_0^1 x^2 dx \text{ bzw. } \int\limits_0^1 x^2 dx$$

⁷Leider geht dieser Gedanke in der Schule immer mehr verloren. Der wesentliche Erkenntnissschritt des Infinitesimalen taucht im Lehrplan nur noch am Rande auf. Die grafikfähigen Taschenrechner, die inzwischen fast überall Einzug gehalten haben, reduzieren viele Probleme der Infinitesimalrechnung auf einfaches Tastendrücken. Der gewaltige Denkfortschritt der Menschheit zu Zeiten von Leibniz und Newton dringt gar nicht mehr in das Denken der Schüler ein.

Oder wie es Eckard K. gesagt hat: Jetzt haben sie die reellen Zahlen wieder abgeschafft.

⁸Für Schüler ist das manchmal ein Problem: die Variable *x* sieht in L^AT_EX so aus: *x*; und das sieht gar nicht so aus wie das *x* aus dem Schreibunterricht der Grundschule.

⁹Beachte das Leerzeichen in der `mbox`-Klammer; ohne das wird das *b* von *berechnet* direkt an das *x* angehängt

6.2.3. Gestapelte mathematische Ausdrücke

Sehr häufig benötigt man in der Mathematik die Möglichkeit, über oder unter mathematische Symbole etwas zu setzen. Die einfachste Anwendung ist, über einen einzelnen Buchstaben ein Unterscheidungszeichen zu machen. Einfallsreich, wie Mathematiker sind, heißt eine Funktion, die sich von einer anderen Funktion f nur wenig unterscheidet, \tilde{f} , die nächste \hat{f} und dann vielleicht \check{f} .¹⁰ Zu diesem Zweck gibt es verschiedene mathematische Akzente, von denen hier `\tildedef`, `\hatf` und `\checkkf` verwendet wurden.

Die nächste Anwendung ist die Unter- oder Überklammerung von Termen, die oft zur Kommentierung oder genaueren Beschreibung eines Ausdrucks benutzt wird. Man findet etwa

$$\underbrace{a_1 + a_2 + \dots + a_n}_{n \text{ Summanden}}$$

oder

$$\overbrace{b_1 \cdot b_2 \cdot \dots \cdot b_k}^{k \text{ Faktoren}}$$

Auch wenn die Befehle dazu, das `underbrace` und das `overbrace` für sich selbst sprechen, ist das hier Geschriebene nicht trivial zu setzen. Das erste Beispiel wird in \LaTeX durch

`\underbrace{a_1 + a_2 + \dots + a_n}_{\mbox{n Summanden}}` erreicht.

Das, was unterklammert werden soll, steht im Parameter für das `underbrace`, und darunter wird durch den Unterstrich tiefergestellt der Text ‚n Summanden‘.

6.2.4. Gleichungen

Bei Gleichungen kommt es vor, dass sie nicht in eine Druckzeile passen, oder dass es gewünscht ist, weil mehrere Gleichheitszeichen auftreten, dass mehrere Zeilen so gesetzt werden, dass diese Gleichheitszeichen untereinander stehen. Dazu gibt es in \LaTeX wieder passende Umgebungen.

Die erste dieser Umgebungen ist das `eqnarray`, also das Feld von Gleichungen. Wie bei einigen anderen Umgebungen existieren hier zwei Varianten: die `eqnarray` und das `eqnarray*`. Bei der `*`-Form werden die einzelnen Zeilen der Gleichung durchnummeriert, und zwar innerhalb eines Kapitels.

Mittels `eqnarray` werden Gleichungen ähnlich gesetzt wie Tabellen. Jede Zeile der Gleichung wird durch einen doppelten Backslash `\\` abgeschlossen, linke und rechte Seite der Gleichung sowie das Gleichheitszeichen werden durch ein Ampersand `&` voneinander getrennt. Als Beispiel werden hier die binomischen Formeln gesetzt:

$$(a + b)^2 = a^2 + 2ab + b^2 \quad (6.1)$$

$$(a - b)^2 = a^2 - 2ab + b^2 \quad (6.2)$$

$$(a + b) \cdot (a - b) = a^2 - ab + ab - b^2 \quad (6.3)$$

$$= a^2 - b^2 \quad (6.4)$$

¹⁰Denn bei Funktionsnamen fängt für Mathematiker das Alphabet bei ‚f‘ an und endet vor ‚g‘.

6. Mathematik (und ein bißchen mehr)

So sieht das aber noch nicht gut aus, denn die letzten beiden Zeilen gehören zur selben Gleichung, sollten also nur eine Nummer bekommen. Also so:

$$(a + b)^2 = a^2 + 2ab + b^2 \quad (6.5)$$

$$(a - b)^2 = a^2 - 2ab + b^2 \quad (6.6)$$

$$\begin{aligned} (a + b) \cdot (a - b) &= a^2 - ab + ab - b^2 \\ &= a^2 - b^2 \end{aligned} \quad (6.7)$$

Das wird durch folgende Eingabe erreicht (Du darfst aber nur einmal raten, durch welchen Befehl ich die Numerierung in der 3. Zeile unterdrückt habe!).

```
1 \begin{eqnarray}
2   (a + b)^2 & & & \& = \& a^2 + 2ab + b^2 \ \\\
3   (a - b)^2 & & & \& = \& a^2 - 2ab + b^2 \ \\\
4   (a + b) \ \mathbf{\cdot} \ (a - b) & \& = \& a^2 - ab + ab - b^2 \ \mathbf{\nonumber} \ \\\
5 & & & \& = \& a^2 - b^2 \\
6 \end{eqnarray}
```

Listing 6.2: Equation-Umgebung

Für Matrizen, lineare Gleichungssystem, Determinanten usw. gibt es eigene Umgebungen, auf die weiter hinten auf Seite 69 eingegangen wird.

6.2.5. Sätze und ähnliches

...nicht deutsche Sätze oder Sätze in einer anderen natürlichen Sprache, sondern Lehrsätze, also Theoreme, Korollare, Lemmata, Definitionen und ähnliches, die in diesem Unterkapitel vereinfachend als „Sätze“ bezeichnet werden. Für diese existiert die Möglichkeit, neue Umgebungen zu schaffen. Der Befehl dazu ist `\newtheorem`.

Um alle Möglichkeiten für die Generierung neuer Satz-Umgebungen zu nutzen, empfiehlt es sich, das Paket `amsthm` mittels `\usepackage{amsthm}` einzubinden. Danach kann man im Vorspann die neuen Umgebungen definieren. Allgemein lautet die Syntax des Befehls `\newtheorem{ndu}[zbx]{vds}[ugz]`. Der erste Parameter `ndu` ist der Name der neuen Umgebung, für die auch gleichzeitig ein neuer Zähler eingerichtet wird. Der zweite Parameter `zbx` (der erste optionale) gibt an, innerhalb welcher Umgebung gezählt werden soll, falls das gewünscht ist. Der dritte Parameter `vds` ist der führende Text, der beim Setzen dieser Umgebung dem Satz vorangestellt wird. Der vierte Parameter `ugz` (der zweite optionale) gibt die Hierarchiestufe der Gliederung des Dokuments an, innerhalb derer gezählt werden soll. Das könnte so aussehen:

```

1 \newtheorem{thm}{Satz}[section] % damit wird die Umgebung
2   % "thm" erzeugt, die Sätze innerhalb von
3   % jeder section durchnummeriert
4 \newtheorem{def}[thm]{Definition} % jetzt bekommt hier die
5   % Umgebung "Def" die Umgebung "thm" als
6   % Eltern-Umgebung, und damit wird jetzt
7   % hier einfach weitergezählt nach dem Satz.
8 \newtheorem{corollary}[thm]{Hilfslemma}
9 \newtheorem{lemma}[thm]{Lemma}
10 \theoremstyle{bemstil}
11 \newtheorem{bem}[thm]{Bemerkung}

```

Listing 6.3: Mathematische Sätze

Der Befehl kann mit bis zu 3 Parametern aufgerufen werden, wobei sich die beiden optionalen Parameter gegenseitig ausschließen.

Im obigen Beispiel wird im ersten Befehl eine Umgebung `thm` geschaffen und gleichzeitig ein Zähler für diese Umgebung eingerichtet. Der zweite Parameter wird hier nicht benutzt, der dritte Parameter setzt als Vortext in dieser Umgebung das Wort „Satz“. Der vierte Parameter gibt hier an, dass innerhalb jeder Section gezählt werden soll. Das heißt z.B., dass in der Section 2 der erste „Satz“ den Zählerwert 2.1 erhält.

Im zweiten Befehl des obigen Beispiels wird die Umgebung für Definitionen erzeugt. Hier ist der zweite Parameter gesetzt, nämlich mittels der Umgebung `thm`. Damit wird erreicht, dass in dem Fall, dass die erste Definition der Section 2, die direkt nach dem (im vorangehenden Abschnitt geschilderten fiktiven) ersten Satz der Section 2 steht, den Zählerwert 2.2 erhält.

6. Mathematik (und ein bißchen mehr)

In der vorletzten Zeile des obigen Beispiels wird ein neuer „Theoremstil“ für die Umgebung `bem` benutzt. Dieser neue Stil wurde durch folgende Definition bereitgestellt:

```
1 \newtheoremstyle{bemstil}% Name des Stils
2 {0.5cm}% Platz oberhalb der Umgebung
3 {0.5cm}% Platz unterhalb der Umgebung
4 {}% Schriftart im Umgebungskeoerper
5 {}% Einrueckung
6 {\normalfont\bfseries}% Schriftart im Umgebungskopf
7 {:}% Zeichensetzung im Umgebungskopf
8 {0.5em}% Abstand nach dem Umgebungskopf
9 {}% händisch gesetzter Umgebungskopf
```

Listing 6.4: Neuer Theoremstil

Das ist ein bißchen lästig, da immer alle Parameter angegeben werden müssen, wobei die Parameter, die gegenüber dem Standard nicht verändert werden sollen, einfach leer bleiben. Im obigen Beispiel wurde für die Umgebung `bem` gewünscht, dass

- oberhalb und unterhalb einer Bemerkung ein vertikaler Leerraum von 5 mm gelassen werden soll
- der Vortext einer Bemerkung fett gedruckt werden soll
- der Umgebungskörper (also der Text der Bemerkung) **nicht** in der sonst bei diesen Umgebungen üblichen Kursivschrift gesetzt werden soll
- nach dem Umgebungskopf ein horizontaler Leerraum von 5 mm gelassen werden soll

Eine weitere Umgebung, die durch das Paket `amsthm` geliefert wird, ist die `proof`-Umgebung für Beweise. Die Darstellung eines Beweises ist folgendermaßen: Das Wort „Beweis“ wird automatisch an den Anfang des Beweises in kursiver Schrift gesetzt, der Beweis wird in normaler Schrift gesetzt. Das Beweisende-Symbol (das quadratische Kästchen) wird automatisch an den rechten Rand der letzten Zeile des Beweises gesetzt. Ein Beispiel dazu folgt:

Satz 6.2.1. *In jede Hosentasche passen unendlich viele Papiertaschentücher.*

Beweis. Vollständige Induktion über n . Induktionsanfang: Für $n = 1$ ist der Satz klar: ein Papiertaschentuch passt in jede Hosentasche. Induktionsschluß: Sei der Satz gültig für n . Es ist ein leichtes zu zeigen, dass ein Papiertaschentuch mehr immer noch in die Hosentasche geht, der Satz also für $n + 1$ gültig ist. \square

Bemerkung 6.2.2: Für den menschlichen Körper und die Halbe Bier gilt der analoge Satz nicht. Vor allem ist von einem Beweisversuch mittels vollständiger Induktion abzuraten.

6.3. Erweiterte Mathematik

Eine weitere schöne Eigenschaft von $\text{T}_{\text{E}}\text{X}/\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ist, dass es für fast alle mathematischen Symbole einen Befehl gibt. Einige der hier aufgelisteten Symbole sind in den Paketen der „American Mathematical Society“ (AMS) enthalten. Deswegen empfiehlt es sich, wenn man Mathematik setzen will, diese Pakete bei mathematischen Texten immer zu laden. Dies geschieht durch die Befehle `\usepackage{amsmath}`, `\usepackage{amsfonts}` und `\usepackage{amssymb}`. Aus diesem Grund stehen die folgenden Eigenschaften von $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ hier in einem eigenen Kapitel.

6.3.1. Pfeile

Sehr häufig benötigt man etwa in mathematischen Formeln Pfeile jedwelcher Art. Hier unten steht eine Tabelle der häufigsten Pfeile und wie man sie in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ schreibt.

Schreibweise	Ergebnis	Schreibweise	Ergebnis
<code>\rightarrow</code>	\rightarrow	<code>\leftarrow</code>	\leftarrow
<code>\Rrightarrow</code>	\Rightarrow	<code>\Lleftarrow</code>	\Leftarrow
<code>\leftrightarrow</code>	\leftrightarrow	<code>\Leftrightarrow</code>	\Leftrightarrow
<code>\longrightarrow</code>	\longrightarrow	<code>\longleftarrow</code>	\longleftarrow
<code>\leftrightharrow</code>	\leftrightarrow	<code>\Leftrightarrow</code>	\Leftrightarrow

Tabelle 6.1.: Mathematische Pfeile

6.3.2. Griechische Buchstaben und andere Symbole

Griechische Buchstaben werden auch so geschrieben, wie sie gesprochen werden.

Schreibweise	Ergebnis	Schreibweise	Ergebnis
<code>\alpha</code>	α	<code>\Omega</code>	Ω
<code>\beta</code>	β	<code>\Lambda</code>	Λ
<code>\gamma</code>	γ	<code>\Gamma</code>	Γ
<code>\delta</code>	δ	<code>\Delta</code>	Δ
<code>\epsilon</code>	ϵ	<code>\Sigma</code>	Σ

Tabelle 6.2.: Griechische Buchstaben

Wichtig



Aber Vorsicht: nicht alle griechischen Buchstaben sind im Mathematik-Modus vorhanden.

6. Mathematik (und ein bißchen mehr)

Andere, seltener benutzte Sonderzeichen kommen aus anderen Schriften:

Schreibweise	Ergebnis	Bedeutung
<code>\aleph</code>	\aleph	
<code>\Re</code>	\Re	für den Realteil einer komplexen Zahl
<code>\Im</code>	\Im	für den Imaginärteil einer komplexen Zahl

Tabelle 6.3.: Andere Buchstaben

Vergleichsoperatoren, so weit das Auge reicht:

Schreibweise	Ergebnis	Schreibweise	Ergebnis
<code>></code>	$>$	<code><</code>	$<$
<code>\geq</code>	\geq	<code>\leq</code>	\leq
<code>\approx</code>	\approx	<code>\approxeq</code>	\approx
<code>\cong</code>	\cong	<code>\equiv</code>	\equiv
<code>\sim</code>	\sim	<code>\neq</code>	\neq

Tabelle 6.4.: Vergleichsoperatoren

Beachte



Wie bei dem letzten Beispiel zu sehen ist, kann ein Vergleichsoperator einfach dadurch verneint werden, dass ein „n“ vorangestellt wird.

Dies sind nicht alle verfügbaren Vergleichsoperatoren, aber hoffentlich die wichtigsten. Falls Dir einer fehlt, schau in der Dokumentation der AMS nach.

Auch Symbole aus der Mengenlehre stehen zur Verfügung:

Schreibweise	Ergebnis	Schreibweise	Ergebnis
<code>\cap</code>	\cap	<code>\cup</code>	\cup
<code>\sqcap</code>	\sqcap	<code>\sqcup</code>	\sqcup
<code>\subset</code>	\subset	<code>\supset</code>	\supset
<code>\subseteq</code>	\subseteq	<code>\supseteq</code>	\supseteq
<code>\in</code>	\in	<code>\ni</code>	\ni
<code>\notin</code>	\notin		
<code>\vartriangleright</code>	\vartriangleright	<code>\vartriangleleft</code>	\vartriangleleft

Tabelle 6.5.: Mengensymbole

6.3.3. Zahlenmengen

Für Mengenbezeichnungen, speziell für Zahlenmengen, nimmt man Großbuchstaben mit Doppelbalken. Die werden so gesetzt:

Schreibweise	Ergebnis	Bedeutung
\mathbb{N}	\mathbb{N}	Menge der natürlichen Zahlen
\mathbb{Z}	\mathbb{Z}	Menge der ganzen Zahlen
\mathbb{Q}	\mathbb{Q}	Menge der rationalen Zahlen
\mathbb{R}	\mathbb{R}	Menge der reellen Zahlen
\mathbb{C}	\mathbb{C}	Menge der komplexen Zahlen
\mathbb{A}	\mathbb{A}	irgendeine Menge

Tabelle 6.6.: Mengenbezeichnungen

6.3.4. Matrizen und ähnliches

Für Matrizen gibt es verschiedene Umgebungen. Hier anschließend wird die Darstellung in \LaTeX gezeigt.

<pre> $\begin{pmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{pmatrix}$ </pre>	$\begin{pmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{pmatrix}$
<pre> $\begin{bmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{bmatrix}$ </pre>	$\begin{bmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{bmatrix}$
<pre> $\begin{Bmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{Bmatrix}$ </pre>	$\begin{Bmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{Bmatrix}$
<pre> $\begin{vmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{vmatrix}$ </pre>	$\begin{vmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{vmatrix}$
<pre> $\begin{Vmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{Vmatrix}$ </pre>	$\begin{Vmatrix} \cos \frac{\pi}{2} & -\sin \frac{\pi}{2} \\ \sin \frac{\pi}{2} & \cos \frac{\pi}{2} \end{Vmatrix}$

Nur die erste Form, mit `pmatrix` gesetzt, ist die „klassische“ Matrix-Form. Aber auch für die anderen Formen gibt es Anwendungen

6.3.5. Kommutative Diagramme

Kommutative Diagramme mit AMSmath

Es gibt mehrere Methoden, um kommutative Diagramme zu erstellen. Die einfachste Art geschieht nur über Befehle der *ams*-Module. Diese kommutativen Diagramme haben auch Einschränkungen in der Komplexität und der Darstellung. Außerdem ist die Schreibweise der Diagramme nicht sehr leser-freundlich. Als Beispiel wird hier ein einfaches quadratisches Diagramm gezeigt:

$$\begin{array}{ccc} x & \xrightarrow{\ln} & \ln x \\ \downarrow \text{exp} & & \downarrow \text{exp} \\ e^x & \xrightarrow{\ln} & x \end{array}$$

Es wurde erstellt durch

```

1      $\begin{CD}
2          x @>\ln>> \ln x\\
3          @VV\mbox{exp}V @VV\mbox{exp}V\\
4          e^x @>\ln>> x
5      \end{CD}$

```

Listing 6.5: Kommutatives Diagramm mit AMSmath

Kommutative Diagramme mit diagrams

Ein weiteres Paket ist „pb-diagram“. Hier muss zuerst wieder das Paket eingebunden werden.

```
1 \usepackage{pb-diagrams}
```

Listing 6.6: Einbinden des Pakets

$$\begin{array}{ccc}
 x & \xrightarrow{\ln} & \ln x \\
 e \downarrow & & \downarrow \ln \\
 e^x & \xrightarrow{\ln} & x
 \end{array}$$

Das obige Diagramm sieht hier so aus:

und wird erzeugt durch

```
1 \node{x} \arrow[e,t]{\ln} \node{\ln x} \arrow[s,l]{e}
2 \node{e^x} \arrow[e,b]{\ln} \node{x} \\
3 \end{diagram}
```

Listing 6.7: Kommutatives Diagramm mit pb-diagram

6.3.6. Stochastik

Okay, die Überschrift ist ziemlich frech, denn ich will hier nur ein schönes Paketchen zeigen, mit dem Würfel gesetzt werden, und die sind ja ganz geschickt für Übungen in Stochastik. So sieht das Mäxle aus: $\square \square$

Dafür bindet man das Paket `epsdice` ein, ein Würfel wird gesetzt mit `\epsdice{zahl}`.

6.3.7. Chemie

Für chemische Reaktionsgleichungen: `\usepackage[version=43]{mhchem}` Dann Formel z.B. `\ce{2H2+O2->2H2O}`

7. Bilder

7.1. Gleitobjekte

Bevor Bilder in L^AT_EX-Dokumente eingebunden werden, muss erst einmal eine Besonderheit hervorgehoben werden. Ein Bild ist ein Objekt, das nicht geteilt werden kann. Es geht nicht, die obere Hälfte des Bildes auf Seite 17 und die untere Hälfte erst auf Seite 18 zu drucken. Ein Bild als ganzes muss an passender Stelle eingebunden werden, und ein Bild braucht in der Vertikalen mehr Platz als nur eine Zeile (fast immer jedenfalls!).

Bilder können fließen! Das bedeutet, dass ein Bild (aber auch andere Objekte) nicht unbedingt an die Stelle gesetzt wird, an der es im Quelltext steht. Das scheint auf den ersten Blick nicht das zu sein, was man wünscht. Aber da muss ein Buchsetzer die Freiheit erhalten, das so zu setzen, wie er es gelernt hat, also soll das auch L^AT_EX, denn L^AT_EX ist ein Buchsatzprogramm (nicht vergessen!!).

Nehmen wir als Beispiel ein Bild, das Platz in der Höhe von 8 Zeilen benötigt; auf der aktuellen Seite sind aber nur noch 5 Zeilen frei. Soll dann mit Gewalt das Bild auf diese Seite gesetzt werden, also mit dem Effekt, dass das Bild dann 3 Zeilen weiter unten endet als der Text auf den umgebenden Seiten? Oder sollen die 5 Zeilen frei gelassen werden und das Bild auf der nächsten Seite ganz oben plaziert werden? Beides sieht nicht gut aus! L^AT_EX muss also selber berechnen, wo Platz für das Bild ist und wo es für eine gleichmäßig aufgeteilte Seite sorgt. Das kann bedeuten, dass das Bild weiter nach oben auf der aktuellen Seite rutscht, aber auch, dass das Bild auf die nächste Seite verschoben wird und Text nach vorne gezogen wird.

Was hier über Bilder gesagt wurde gilt auch für die anderen typischen Vertreter von Gleitobjekten, die Tabellen. Auch diese Objekte sollen als ein Stück behandelt werden, also zusammenbleiben und auf eine Seite gesetzt werden. Bei Tabellen ist das allerdings manchmal nicht möglich, während bei Bildern das unumgänglich ist. Zur Not muss ein Bild so verkleinert werden, dass es auf eine Seite passt. Bei einer Tabelle ist es aber durchaus möglich und vertretbar, sie über mehrere Seiten zu verteilen.¹

Aus diesem Grund gibt es für Bilder und Tabellen Gleit-Umgebungen, nämlich `figure` und `table`.

¹ein schönes Beispiel dazu ist das statistische Jahrbuch der Bundesrepublik Deutschland: Tabellen über Tabellen, eine schöner und länger als die andere

7.2. Früher und heute

Als \LaTeX jung war, war das Standard-Ausgabeformat Postscript. Aus diesem Grund war es früher auch nur möglich, als Bilder eps-Dateien² einzubinden. Seither hat sich sowohl bei Dokumentenformaten und Bildformaten einiges geändert. Heute ist das Standard-Ausgabeformat pdf, und Bilder werden meistens als jpg-³ oder als png-Dateien⁴ abgelegt.

7.2.1. Das Einbinden von Bilddateien

Zuerst muss natürlich wieder ein Paket geladen werden. Hier ist ein kleiner Fallstrick, wenn man nicht aufpasst. Das Paket, das für Bilder geladen wird, heißt `graphicx`, wird also mit `\usepackage{graphicx}` geladen.

Die Bilder selber werden eingebunden mit dem Befehl

`\includegraphics{Bilddatei.name}` ... und das Ergebnis sieht meistens grauenvoll aus. Denn meistens passt das Bild von der Größe nicht zum Text. Aus diesem Grund hat der obige Befehl noch einen optionalen Parameter, der für die Größe zuständig ist, nämlich

`\includegraphics[scale=?]{Bilddatei.name}` oder

`\includegraphics[width=?]{Bilddatei.name}`.

Damit gibt es insgesamt 3 Möglichkeiten der Skalierung eines Bildes.

1. Es kann eine absolute Größe — genau gesagt: eine absolute Breite — für das Bild angegeben werden mit `\includegraphics[width=8cm]{Bilddatei.name}`, das Bild wird also auf eine Breite von 8 Zentimetern zurechtgeschnitten.
2. Das Bild kann relativ zur Originalgröße skaliert werden. `\includegraphics[scale=0.5]{Bilddatei.name}` verkleinert also das Bild auf halbe Größe
3. Oder das Bild wird relativ zur Textzeile skaliert durch `\includegraphics[width=.3\textwidth]{Bilddatei.name}`, womit das Bild auf 30% der Breite des Textes zurechtgeschnitten wird.

7.2.2. Das Drumherum

Ein Abbildungsverzeichnis, wenn man denn wirklich mehrere Abbildungen in seinem Werk hat, ist eine feine Sache. Mit \LaTeX wird es durch den Befehl `\listoffigures` gesetzt, normalerweise an das Ende des Werks.

Wie schon oben erwähnt, muss ein Bild manchmal gleiten. Dazu gibt es die `figure`-Umgebung. Diese Umgebung versucht, ein Bild optimal zu platzieren. Dabei kann als

²siehe Encapsulated Postscript (EPS)

³siehe Joint Photographic Experts Group (JPG)

⁴siehe Portable Network Graphics (PNG)

optionaler Parameter eine Empfehlung mitgegeben werden, in der der Schreiber vorschlägt, wo das Bild wenn möglich zu setzen ist. Die Befehlsfolge für ein solches gleitendes Bild ist also so:

```

1 \begin{figure}[htb]
2 \includegraphics[scale=0.5]{martin.jpg}
3 \end{figure}

```

Listing 7.1: Figure-Umgebung

Hier werden im optionalen Parameter für die Positionierung drei mögliche Werte angegeben. Die Werte bedeuten im einzelnen:

- h** here. Das Gleitobjekt soll an der Stelle gesetzt werden, wo es im Quelltext steht.
- t** top. Das Gleitobjekt soll an den Kopf der laufenden Seite gesetzt werden, wenn der Textfluss es erlaubt. Wenn nicht soll es an den Kopf der folgenden Seite gesetzt werden.
- b** bottom. Das Gleitobjekt soll an den Fuß der laufenden Seite gesetzt werden, wenn der Textfluss es erlaubt. Wenn das nicht passt, soll es an den Fuß der folgenden Seite gesetzt werden.
- p** page of floats. Gleitobjekte werden auf eine Extra-Seite gesetzt.

Die obige Parameterkombination gibt also eine Präferenz-Reihenfolge an, die \LaTeX auffordert, zu probieren, ob das Objekt „here“ gesetzt werden kann, wenn das nicht möglich ist, soll \LaTeX es „top“ zu setzen versuchen, und wenn das auch nicht geht, „bottom“.

Zusätzlich erlaubt die `figure`-Umgebung, eine Bildüberschrift bzw. eine Bildunterschrift zu setzen. Dies geschieht durch den Befehl `\caption{Das zeigt das Bild}`. Eine Bildüberschrift wird also gesetzt durch

```

1 \begin{figure}[htb]
2 \caption{Das zeigt das Bild}
3 \includegraphics[scale=0.5]{martin.jpg}
4 \end{figure}

```

Listing 7.2: Bildüberschrift

eine Bildunterschrift durch

```

1 \begin{figure}[htb]
2 \includegraphics[scale=0.5]{martin.jpg}
3 \caption{Das zeigt das Bild}
4 \end{figure}

```

Listing 7.3: Bildunterschrift

7. Bilder

7.2.3. Absolute Positionierung

Es ist durch die Einbindung eines weiteren Pakets möglich, eine absolute Positionierung vorzunehmen, nämlich durch `\usepackage{float}`. Die wichtigste Verbesserung durch dieses Paket ist die Option „H“ als optionalen Parameter für eine Positionierung genau an der Stelle, wo das Objekt im Quelltext auftaucht.

Gerade bei Lehrbüchern muss das oft so sein. Aber man sollte sich im Klaren darüber sein, dass das oft nicht zum schönen Aussehen beiträgt. Manchmal bleiben durch diese absolute Positionierung halbe Seiten leer, und das bedeutet, dass man oft von Hand nachbessern muss, unter Umständen dadurch, dass man Text streicht oder stretcht.

7.3. Fließtext um ein Bild

7.3.1. floatingfigure

Das lässt einen Text mit Bildern erst richtig professionell wirken! Wenn ein Bild in einem Text nicht eine eigenständige Aussage macht, sondern eine Ergänzung des Textes darstellt, dann sollte das Bild möglichst von Text umflossen werden. In speziell für solche Aufgaben designeten Programmen geschieht das manuell, indem man an Bild und Text „Anfasser“ bedienen kann, die Bildgröße und Texthöhe verändern. Das funktioniert wirklich gar nicht schlecht, auch wenn dabei Augenmaß, Probieren und Geduld nötig sind. Für ein Satzsystem wie \LaTeX ist das natürlich unter dem eigenen Anspruch: die Platzierungen, die Verteilung von Text und Bild, all das kann \LaTeX selber ausrechnen!

Dieses Bild im vorigen Absatz wurde mit der Umgebung `floatflt` gesetzt. Dazu muss im Vorspann diese Umgebung eingebunden werden, was mit dem Befehl `\usepackage{floatflt}` geschieht.

Ein Bild steht dann in einer `floatingfigure`-Umgebung. Das obige Bild wurde zum Beispiel gesetzt mit

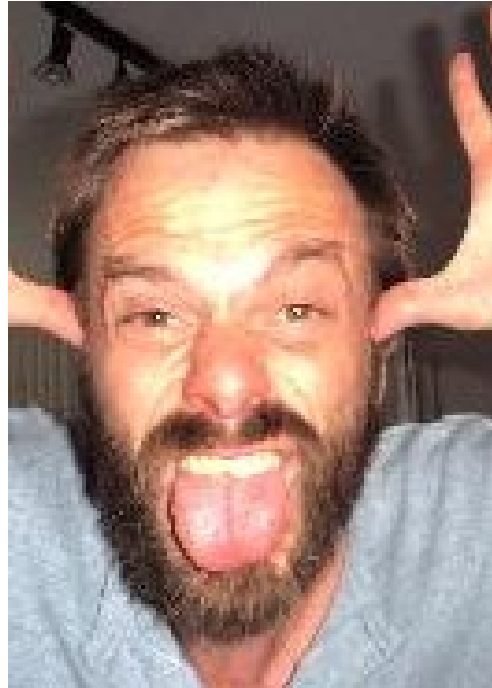


Abbildung 7.1.: Vanitas vanitatis ...

```

1  \begin{floatingfigure} {7.5cm}
2  \mbox{\includegraphics [width=6.5cm] {../Bilder/martin.jpg}}
3  \caption{Vanitas vanitatis \ldots}
4  \end{floatingfigure}

```

Listing 7.4: Floatfigure-Umgebung

Die Breite des Bildes wird dabei als Parameter an die Umgebung weitergegeben. Wie bei anderen Gleitobjekten auch, kann hier eine Bildüber/-unterschrift mit Hilfe von `\caption{}` gesetzt werden, die dann auch in ein Abbildungsverzeichnis übernommen wird. In diesem Fall wurde das Bild in eine `\mbox{}` eingefügt.

`floatingfigure` setzt Bilder immer auf die selbe Art: an den rechten Rand auf ungeraden Seiten, auf den linken Rand auf geraden Seiten. Wie man bei dem obigen

7. Bilder

Beispiel aber auch sieht: der rechte Rand zerfleddert, Silbentrennung funktioniert nicht richtig, das ganze ist nicht sehr zufriedenstellend.

7.3.2. wrapfigure

Ein weiteres Paket, das Fließtext bearbeitet, ist `wrapfig`. Es wird mit `\usepackage{wrapfig}` eingebunden. Das wurde mit dem nebenstehenden Bild gemacht. Der erste Unterschied ist sofort zu sehen: auch dieses Bild steht am rechten Rand der Seite, rechts des Textes, obwohl es sich auf einer geraden Seite befindet. Hier kann man also über einen weiteren Parameter die Positionierung des Bildes im Verhältnis zum Text bestimmen.

Hier, wie natürlich auch in der `floatingfigure`-Umgebung, ist es angebracht die Skalierung des Bildes auf die selbe Art und Weise zu machen wie die Skalierung der Umgebung. Wenn ich also wie in diesem Beispiel eine absolute Breite der Umgebung durch

`\begin{wrapfigure}{r}{7cm}` vorschreibe, ist es gut, die Größe des Bildes auch absolut anzugeben.

Es wird selbstverständlich so viel Text benutzt, wie nötig ist, um den gesamten Raum neben dem Bild auszufüllen. Erst danach wird wieder über die gesamte Breite des Blattes geschrieben. (Und dieser Absatz wurde nur deswegen gemacht, damit man diesen Sachverhalt gut sieht!!)

Das obige Beispiel wurde gesetzt mit

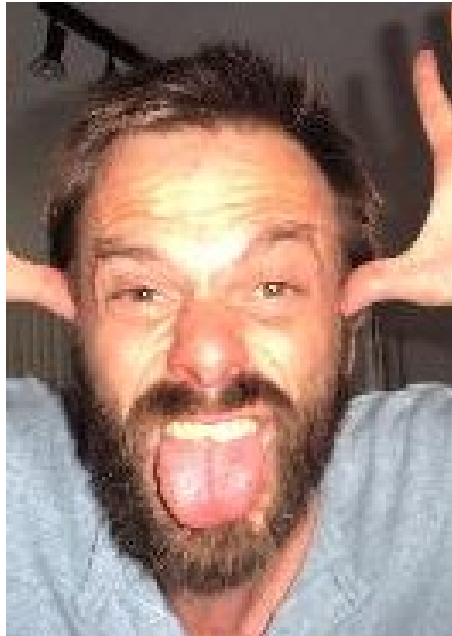


Abbildung 7.2.: ... et omnia vanitas

```

1 \begin{wrapfigure}{r}{7cm}
2 \mbox{\includegraphics[width=6cm]{../Bilder/martin.jpg}}
3 \caption{\ldots et omnia vanitas }
4 \end{wrapfigure}

```

Listing 7.5: Wrapfigure-Umgebung

7.4. Teilbilder

(subfigure)

7.5. Spielereien

7.5.1. Ziertexte

Für manche Texte will der Autor eine andere Form als das Rechteck haben. Ein nettes Beispiel dafür ist der Roman „Verteidigung der Missionarsstellung“ von Wolf Haas, der Seiten enthält, die nicht dem entsprechen, was wir von dem Füllen einer Seite gewohnt sind.

Die erste Möglichkeit ist die Verwendung des Paketes `shapepar`. Dieses stellt die Formen Herz, Karo und Quadrat bereit. Diese Formen werden benutzt durch die Befehle `\heartpar{}Text{}` bzw. `\diamondpar{}Text{}` bzw. `\squarepar{}Text{}` .

Die Ergebnisse sind dann

Das kann man
natürlich für einen Liebes-
brief benutzen, obwohl jeder mit-
telmäßig romantische Mensch hier doch
lieber zu Papier und Stift, vorzugsweise
sogar zu einem Füllfederhalter oder so-
gar zu Feder und Tinte greifen wird.

Auch ein Tintenklecks in einem
solchen Werk zeugt nur von
der tiefen emotionalen
Ergriffenheit des
Schreibers.



oder



Das
deutsche
Kartenspiel
Skat wird mit
einem französischen
Blatt, das folglich die Spiel-
farben Kreuz, Pik, Herz und
Karo enthält, gespielt. In jeder
Farbe sind die Werte As,
Zehn, König, Dame,
Bube, Neun, Acht
und Sieben
vorhan-
den.



oder

7.5. Spielereien

Was fällt mir doch gleich
bei quadratisch ein?
Praktisch und gut, und
dass ich mal wieder ei-
ne Runde mit dem Fahr-
rad durch den Schönbuch
drehen könnte. Vielleicht
hat der Fabrikverkauf auf
und es gibt wieder Bruch.

Text entlang einer beliebigen
Kurve, hier einer Sinus-Kurve

8. Fußnoten und Randnoten

8.1. Fußnoten

Fußnoten zu setzen ist eines der einfachsten Dinge in \LaTeX . An der Stelle, an der ein Verweis auf eine Fußnote gemacht werden soll, wird der Befehl `\footnote{}` angegeben, so wie hier.¹ \LaTeX zählt die Fußnoten mit, berechnet den Platz, der auf einer Seite für Fußnoten gelassen werden muss, und schreibt die Fußnote unter einen schmalen Strich ans Ende der Seite. Und das sieht dann so aus wie hier unten.

Man muss sich wirklich um nichts kümmern. Das nachträglich Einfügen einer weiteren Fußnote an einer beliebigen Stelle des Textes führt nur zu einer Neuberechnung des Fußnoten-Zählers und zu einer Neuberechnung der Verteilung des Textes auf die Seiten. Und beim Berechnen ist der Computer und ist \LaTeX besser und schneller als wir.²

8.1.1. Fußnoten in Überschriften³ ...

... sind eine Frage des Geschmacks! Manche Schriftkünstler lehnen das grundsätzlich ab, aber manche Schreiber benötigen sie. Fußnoten in Überschriften bringen aber \LaTeX ins Schleudern. Deshalb müssen Fußnoten in Überschriften mit Hilfe des Befehls `\protect` geschützt werden. Der Befehl `\protect` wird direkt vor dem zu schützenden Befehl eingefügt. Man schreibt also

```
\section{Fußnoten in Überschriften\protect\footnote{Genau wie Fortsetzungspunkte} \ldots}
```

¹irgendein Fußnotentext

²und auf jeden Fall schneller als ein Mensch mit einem Textverarbeitungsprogramm

³Genau wie Fortsetzungspunkte

8.1.2. Fußnoten im zweispaltigen Satz

Hier wird also ein kleiner Textabschnitt zweispaltig gesetzt. Die Option `twocolumn` im Vorspann erlaubt den zweispaltigen Satz eines ganzen Dokuments. Außerdem kann innerhalb eines Dokuments mit dem Befehl `\twocolumn` auf zweispaltigen Satz umgeschaltet werden. Dabei ist aber zu beachten, dass dadurch immer eine neue Seite begonnen wird. Das ist eine insgesamt nicht wünschenswerte Eigenschaft. Abhilfe schafft das Paket `multicols`, das es ermöglicht, innerhalb einer Seite eine zweispaltige Umgebung

zu schaffen. Aber da wünscht man sich dann eine Fußnote doch vielleicht am unteren Rand des Abschnittes, der in zwei Spalten steht⁴. Dies geschieht durch die Umgebung `multicols`, wird also durch `\begin{multicols}{anzahl Spalten}` eingeleitet und durch `\end{multicols}` abgeschlossen. (Ich gebe zu, diese ganzen Sätze werden künstlich in die Länge gezogen, um einen ernsthaften zweispaltigen Text zu produzieren. Der Stil heiligt die Mittel ... oder so ähnlich.)

Hier soll also eine Fußnote am Ende des zweispaltig gesetzten Textes stehen. Dazu muss man die `\begin{multicols}{2}` in eine `minipage`-Umgebung einbinden, also

```
Text .  
\end{multicols}  
\end{minipage}
```

```
\begin{minipage}{\textwidth}  
  \begin{multicols}{2}  
    bla bla.  
    Hier steht  
    zweispaltiger
```

Wenn jetzt also hier^a eine Fußnote mit `\footnote{}` eingefügt wird, dann bekommt sie eine andere Numerierung und wird unter den zweispaltigen Einschub gestellt.

^aso ist es!

Falls das gesamte Dokument zweispaltig gesetzt wird, wenn also in der `documentclass`-Zeile die Option `twocolumn` gesetzt ist, ist es oft wünschenswert, dass auf jeder Seite die Fußnoten an das Ende der rechten Spalte gestellt werden. Dazu bindet man das Paket `ftnright` ein. Weiter ist nichts zu tun, denn jetzt werden alle Fußnoten so behandelt, wie es im vorigen Satz beschrieben ist. (Das kann ich hier also leider nicht vormachen, denn hier werden nur einzelne Passagen zwei- oder mehrspaltig gesetzt.)

8.1.3. Fußnoten unter Tabellen

Das oben über Fußnoten im zweispaltigen Satz gesagte kann auch auf Fußnoten in Tabellen übertragen werden. Wenn es gewünscht ist, dass innerhalb einer Tabelle Fußnoten auftreten, diese aber nicht am Ende der Seite sondern direkt unter der Tabelle stehen sollen, ist es am einfachsten, die ganze Tabelle in eine `Minipage`-Umgebung einzubinden.

⁴und nicht so, am Ende der Seite

8.2. Randnoten

Randnoten können dazu dienen, während der Arbeit an einem Schriftstück Notizen für die noch fehlende Bearbeitung zu setzen. Es gibt aber durchaus, vor allem in wissenschaftlichen Texten, einige Gründe, mit Randnoten zu arbeiten.

In Lehrbüchern kann zum Beispiel auf eine Besonderheit, ein wichtiges Beispiel oder etwas ähnliches hingewiesen werden. Dies geschieht durch den Befehl `\marginpar{}`. Da \LaTeX Randnoten bei zweiseitigen Dokumenten immer an den äußeren Rand setzt, wird der Pfeil am Rand natürlich auf einer linken Seite am linken Rand stehen und muss daher nach rechts zeigen, auf einer rechten Seite natürlich nach links. Mit \LaTeX ist das kein Problem: dem `\marginpar` kann ein optionales Argument und ein Pflicht-Argument mitgegeben werden, also `\marginpar[links]{rechts}`. Damit wird auf linken Seiten das „links“ und auf rechten Seiten das „rechts auf den linken beziehungsweise rechten Rand gesetzt. Der Pfeil hier wurde mit `\marginpar[\Longrightarrow]{ \Longleftarrow }` gesetzt.

TO DO!

←

8.3. Querverweise

Bei Querverweisen kann auf

- eine Seite oder auf
- einen mit einer Marke versehenen Gliederungsbehl

verwiesen werden.

In beiden Fällen muss die Stelle, auf die verwiesen werden soll, mit einer Marke versehen werden. Dies geschieht durch den \LaTeX -Befehl `\label{meineMarke}`. Die Marke darf aus einer beliebigen Zeichenfolge bestehen (allerdings ohne Sonderzeichen und in \TeX reservierte Zeichen).

Bei der Referenzierung auf eine Seite wird durch den Befehl `\pageref{meineMarke}` an die aktuelle Stelle die Seitennummer der markierten Seite geschrieben. Bei der Referenzierung auf einen Gliederungsbehl wird durch den Befehl `\ref{meineMarke}` an die aktuelle Stelle die Nummer des Gliederungsbehl geschrieben.

8.3.1. Querverweise in pdf-Dokumenten

Seit es Standard geworden ist, \LaTeX -Dokumente nicht mehr in das `.div`-Format sondern in das `.pdf`-Format umzuwandeln, ist es eine besonders komfortable Einrichtung, dass man in diesem Format Hyperlinks einsetzen kann. Hyperlinks sind aus dem Internet bekannt: ein Teil des Textes ist besonders markiert und wird für Maus-Aktionen aktiv, was bedeutet, dass man mit Hilfe eines Mausklicks an die Stelle springt, auf die quer-verwiesen wird.

Dafür ist in \LaTeX nicht viel nötig. Zuerst muss natürlich wieder ein Paket eingebunden werden, das Paket `hyperref`. Eine Referenz auf eine andere Seite (im WWW) wird dann durch einen `\href{hier gehts hin}{http://www.die.adresse}` erreicht. Alle anderen Referenzen sind automatisch in pdf-Dokumenten Hyperlinks.

8. Fußnoten und Randnoten

8.3.2. Querverweise auf andere Gliederungselemente

Binde ein: `\usepackage{pdfpages}` Markierung: `\label{enum:this1}` Verweis: `\ref{enum:this1}`

9. Komplexe Dokumente

9.1. Der erste Eindruck zählt: Die Titelseite

Selbstverständlich unterstützt L^AT_EX in bequemer Weise die Erstellung einer Titelseite. Schließlich kommt es darauf wirklich besonders an, dass man gleich mit der ersten Seite dem Leser (der vielleicht auch gleichzeitig derjenige ist, der die Arbeit bewerten soll) zeigt, dass das Werk etwas besonderes ist. Für die detaillierte Gestaltung von Titelseiten und der Bedeutung, die die einzelnen Seiten haben, die vor den eigentlichen Text gestellt werden, verweise ich auf das Buch von Berndt.¹

Unter L^AT_EX gibt es (wenigstens) zwei Möglichkeiten, Titelseiten zu erstellen: zum einen, indem man mit den Bordmitteln von L^AT_EX ein Aussehen schafft, das den eigenen Vorstellungen entspricht, zum anderen, indem man die für Titelseiten speziell vorgesehenen Befehle benutzt. Im folgenden wird hier nur auf die zweite Methode eingegangen, da diese erstens sehr gute Ergebnisse liefert und da zweitens das eingeständige Erstellen nur die Benutzung von schon bekannten Befehlen in anderem Zusammenhang voraussetzt.

9.1.1. Die Standard-Titelseite

Die Titelseite eines Werkes ist eine Umgebung für sich, die durch `\begin{titlepage}` eingeleitet und durch `\end{titlepage}` beendet wird. Innerhalb dieser Umgebung sind wenige Befehle zulässig. Es sind dies

- `\title`
- `\author`
- `\date`
- `\thanks`

Die Namen der Befehle sagen schon aus, wofür sie benutzt werden. Der Befehl „thanks“ kann dabei an jeder Stelle von Titel, Autor oder Datum stehen. Der Text, der nach dem „thanks“-Befehl steht, wird dann in einer Fußnote eingefügt.

Wird der Befehl „date“ nicht gesetzt, wird automatisch das Datum der Generierung der Ausgabe genommen.

Wenn die Arbeit von mehreren Autoren geschrieben wurde, werden die einzelnen Autoren durch den Befehl `\and` voneinander getrennt. Dabei kann ein einzelner Autor-Eintrag nicht nur aus dem Namen des Autors bestehen, sondern etwa noch durch einen Ort, ein Institut, eine Firma etc. ergänzt werden.

¹berndt-ldtemd2008

9. Komplexe Dokumente

Jetzt kommt aber das wichtigste: wenn man diese Befehle in sein Dokument eingefügt hat, wird noch keine Titelseite gesetzt. Das geschieht erst durch den Befehl `\maketitle`

So sieht also beispielsweise eine Titelseite im Quelltext aus:

```
1 \title{Einführung in \LaTeX}
2 \author{Martin Schimmels and viele andere}
3 \maketitle
```

Listing 9.1: Satz einer Titelseite

Das sieht dann so aus:

Einführung in L^AT_EX

Martin Schimmels viele andere

12. April 2012

1

Abbildung 9.1.: So sieht die Titelseite aus

9.1.2. Die Standard-Titelseite bei Benutzung von KOMA

Für Benutzer von KOMA-Script, der Anpassung für die Belange deutschsprachiger Nutzer, stehen dem Schreiber noch mehr Möglichkeiten zur Verfügung, eine Titelseite zu gestalten. Die einzelnen Befehle werden aufgelistet und anschließend erklärt.

- `\titlehead`
- `\subject`
- `\title`
- `\author`
- `\date`
- `\publishers`
- `\thanks`

Hier kommt also eine Standard-Titelseite mit KOMA-Script.

```
1 \documentclass[10pt,a4paper,ngerman,german,utf8]{scrartcl}
2 \usepackage{babel}
3 \usepackage[utf8]{inputenc}
4 \usepackage{ucs}
5 \usepackage{amsmath}
6 \usepackage{amsfonts}
7 \usepackage{amssymb}
8 \author{Martin Schimmels and Noch Jemand}
9 \title{Titelseite mit KOMA-Script}
10 \subject{Veränderungen der Titelseite(n) unter KOMA}
11 \titlehead{Erstellt zu Rottenburg i.J.d.H. 2011}
12 \publishers{veröffentlicht unter der GFDL}
13 \usepackage{graphics}
14 \begin{document}
15   \maketitle
16   \newpage
17   hier kommt der Text
18 \end{document}
```

Listing 9.2: Satz einer Titelseite mit KOMA

Zeile 8 Wie bei der Standard-Titelseite gilt auch hier, dass mehrere Autoren durch `\and` voneinander getrennt werden.

Zeile 9 Der eigentliche Titel der Arbeit, groß und fett gesetzt.

Zeile 10 Das `\subject`, also das Thema der Arbeit, steht über dem eigentlichen Titel in einer etwas kleineren Schrift.

9.1. Der erste Eindruck zählt: Die Titelseite

Zeile 11 Der `\titlehead` steht noch oberhalb der eigentlichen Titelei in der linken oberen Ecke der Titelseite.

Zeile 12 Ein Attribut für den Verleger (den Verlag), das natürlich auch für andere Informationen gebraucht (oder mißbraucht) werden kann. Der Verleger steht immer als letztes Element auf der Titelseite.

Die oben beschriebene Titelseite mit KOMA sieht so aus:

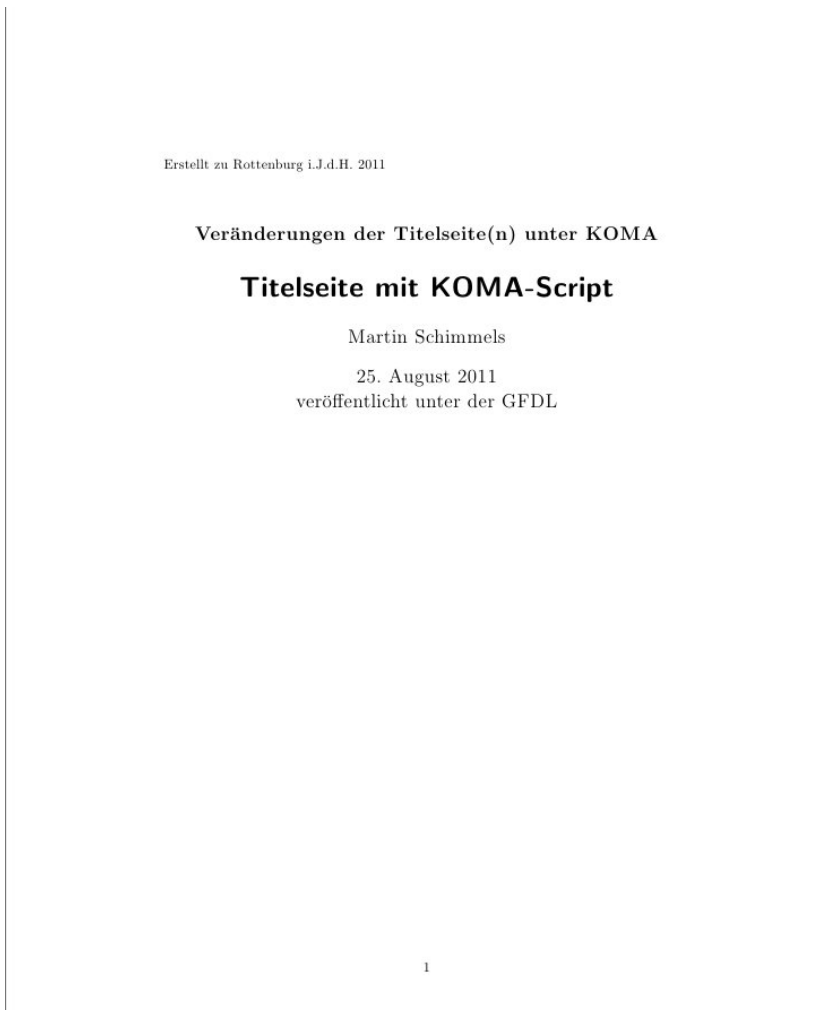


Abbildung 9.2.: So sieht die Titelseite aus

9.2. Mein Prof will ein ganz anderes Aussehen!

Wie schon mehrmals erwähnt: L^AT_EX-Dokumente sehen gut aus! Wenn man an der Darstellung etwas ändert, dann sollte man genau wissen was man tut. Manchmal ist es aber unerlässlich, weil genaue Anforderungen angegeben sind, wie eine Arbeit auszusehen hat.

Auch hier gilt eine Hauptregel: Änderungen, die das Aussehen eines Dokuments als Ganzes betreffen, sollten immer im Vorspann deklariert werden. Nur so kann man sicher gewährleisten, dass eine Einheitlichkeit des endgültigen Produkts herauskommt. Man sollte sich also sehr davor hüten, an einer Stelle des Quelltextes eine Änderung anzubringen.

9.2.1. Eine andere Schrift

Diese Überschrift müsste eigentlich heißen: eine andere Schriftfamilie. Denn hier geht es darum, für das gesamte Dokument, das man erzeugen will, eine andere Standardschrift auszuwählen. Die Standardschrift unter L^AT_EX ist die „cmr“, die Computer Modern Roman. Eine kurze Geschichte dieser Schrift findet man im L^AT_EX-Begleiter² auf Seite 159f.

Schriften werden nach verschiedenen Kriterien eingeteilt. Die erste Einteilung ist die in Proportionalschriften und Schriften mit fester Zeichenbreite.

Proportionalschriften und Nichtproportionalschriften

Die zweite Art von Schriften, die Nichtproportionalschriften, ist typisch für die gute alte Schreibmaschine. Wer sich ein solches Werkzeug einmal angeschaut hat, dem ist klar, warum hier alle Buchstaben, also sowohl das schlanke *i* als auch das breite *m* die selbe Breite im Text haben. Selbst wenn die Hämmer der Schreibmaschine verschiedene Breite haben können, bewirkt jeder Anschlag nicht nur die Bewegung des Hammers zum Papier und zurück, sondern auch ein Vorwärtstransportieren des Wagens. Dies ist mechanisch gelöst, und es wäre für eine solch rein mechanische Maschine eine große Herausforderung, je nach Auswahl der Taste einen spezifischen Vorschub zu machen! Also wird für jeden Anschlag der Wagen um die selbe Distanz vorgeschoben.

Proportionalschriften hingegen haben für verschiedene Buchstaben verschiedene Breiten. Wer einmal einem Drucker über die Schulter geschaut hat oder in einem Museum eine Druckerwerkstatt bewundert hat, dem ist auch das schon aufgefallen: die einzelnen Lettern sind unterschiedlich breit. Das geht sogar noch weiter: die Lettern für einen Buchstaben konnten in geringfügig unterschiedlicher Breite vorliegen, damit der Setzer je nach Bedarf den Zwischenraum zwischen einzelnen Buchstaben vergrößern oder verkleinern konnte, so dass eine Zeile immer exakt gefüllt und trotzdem harmonisch war.

Da zu Beginn der EDV Schreiben auf dem Computer sich am Schreiben auf der Schreibmaschine orientierte, waren die ersten Schriften auf den Computern Nichtproportionalschriften. Das passte auch gut zu den ersten Druckern, die es zu kaufen gab.

²goossens-dl1995.

9.2. Mein Prof will ein ganz anderes Aussehen!

Diese ratternden Nadeldrucker hatten für jedes Zeichen eine in der Größe feststehende Matrix, in die das Zeichen eingepasst wurde. Hier war ein „m“ genau so breit wie ein „i“. Noch in den 70er-Jahren des vorigen Jahrhunderts hat man an Universitäten Lehrbücher gefunden, meistens Vorlesungsmitschriebe, die so aussahen. Mit dem technischen Fortschritt und dem Auftauchen von Tintenstrahldruckern, vor allem aber dem der Laserdrucker, ist diese Einschränkung weggefallen. Heutzutage schreibt niemand mehr mit einer Schreibmaschinenschrift seine wissenschaftliche Arbeit.³

Bei der Verwendung von Nichtproportionalschriften ist auch noch ein anderes Relikt aus längst vergangenen Tagen zu beobachten. Während ein Text in Proportionalschrift auch deswegen schön aussieht, weil er im Blocksatz gesetzt ist, was heißt, dass jede Zeile gleich lang ist und rechtsbündig und linksbündig schließt, sieht das bei einer Nichtproportionalschrift einfach nur bescheiden aus. Nichtproportionalschriften in Blocksatz zu zwingen ist eine ästhetische Beleidigung. Solche Schriften müssen einfach im Flattersatz stehen, und das ist nicht sehr leserfreundlich, denn die Augen des Lesers müssen so beim Zeilenwechsel unterschiedlich große Sprünge machen, und das ermüdet die Augen.

Serifen oder keine Serifen

Serifen sind die kleinen Häkchen an den einzelnen Buchstaben. Diese haben historisch eine Daseinsberechtigung: als Buchsatz noch mit Bleiletern aus einem Setzkasten gemacht wurde, nahm der Setzer die Lettern aus seinem Setzkasten und setzte sie in eine Zeile aus Holz. Damit die Lettern dabei nicht verrutschten und schön an Ort und Stelle blieben, bekamen sie kleine Füßchen um sich festzuhalten, die Serifen.

Obwohl heutzutage serifenlose Schriften fast immer als eleganter angesehen werden, sind viele Bücher in Serifenschriften gesetzt, egal ob Roman oder Sachbuch. Auch das hat seinen guten Grund. Denn die Füßchen, die sich früher an der Setzzeile festgehalten haben, geben jetzt dem Auge des Lesers halt, so dass dieses nicht aus der Zeile springt. Texte in Serifenschriften sind somit weniger ermüdend, weil das Auge eine Unterstützung bekommt.

9.2.2. Hervorhebungen

In \TeX werden Textstellen durch Schrägstellen hervorgehoben. Der Befehl dazu ist das `\em` für „emphasize“. In gedruckten Werken ist es verpönt, Textstellen, die hervorgehoben werden sollen, zu unterstreichen. Auch hier sollte man im Zweifelsfall mit der Person, der Firma oder der Stelle sprechen, für die man die Arbeit schreibt. Nur wenn derjenige oder diejenigen es ausdrücklich wünschen, sollte man Hervorhebungen durch Unterstreichen realisieren.

Sollen mehrere Wörter an einem Stück unterstrichen werden, das heißt mit den Wortzwischenräumen, macht man das durch

```
\underline{diesen ganzen Text unterstreichen}.
```

³... außer Herr Tobias S., der von seinem Prof als Vorgabe für seine Masterarbeit „Schreibmaschinenschrift, 2-zeiliger Abstand“ vorgegeben bekam. Aber das erzählen wir hier nicht weiter!

9. Komplexe Dokumente

Der Fettdruck für Hervorhebungen erregt in einem Vorlesungsskript oft eher die Aufmerksamkeit des Lesers als ein Schrägstellen. Auch das ist in \LaTeX möglich; der Befehl dazu ist `\bfseries` für „boldface“

Im Gegensatz zu Textverarbeitungen, bei denen eine Veränderung der Schriftgröße in absoluten Angaben möglich ist (entweder in Punkt oder in cm), findet man in \LaTeX relative Schriftgrößen. Das bedeutet, dass ausgehend von der Standardschriftgröße im Verhältnis dazu größere und kleinere Schriften über Befehle ausgewählt werden. Der Vorteil liegt wieder auf der Hand: die Größenverhältnisse passen.⁴

Die verschiedenen Schriftgrößen werden hier aufgelistet:

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

9.2.3. Absätze

Die erste Zeile von Absätzen wird standardmäßig eingerückt. Wenn man das für das gesamte Dokument ausschalten will, geschieht das selbstverständlich im Vorspann, und zwar mit dem Befehl `\parindent=0mm`.

Wenn nur einzelne Absätze nicht eingerückt werden sollen, wird vor den entsprechenden Absatz der Befehl `\noindent` gesetzt.

Für längere Zitate, aber auch zur Hervorhebung eines Textabschnittes, soll manchmal ein ganzer Absatz eingerückt werden. Dazu dient die Umgebung `quote`, die folglich mit `\begin{quote}` anfängt und mit `\end{quote}` endet.

Dabei können durchaus mehrere Absätze in einer solchen Umgebung stehen, so dass das dann in der Ausgabe wie dieser und der vorhergehende Absatz erscheint.

⁴Es ist bei der Arbeit mit einer Textverarbeitungen einer der häufigsten Fauxpas, von Hand eine Vergrößerung der Schrift vorzunehmen. Der Experte sieht sofort, dass da ein ästhetischer Bruch ist.

9.2. Mein Prof will ein ganz anderes Aussehen!

Es gibt noch eine zweite Umgebung für das Einrücken von Absätzen, die `quotation`-Umgebung. Der Unterschied zur obigen `quote`-Umgebung ist, dass hier die erste Zeile eines jeden eingerückten Absatzes nochmals eingerückt wird, wie das im normalen Text auch gemacht wird.

Und zur Demonstration wird hier noch ein ziemlich überflüssiger Absatz angehängt, sonst glaubt das niemand, dass die erste Zeile eines Absatzes eingerückt wird. Dafür wird in dieser Umgebung kein zusätzlicher Zwischenraum zwischen die Absätze eingeschoben.

9.2.4. Abstände

Horizontale Abstände werden durch den Befehl `\hspace{Längenangabe}` gemacht. Soll eine Zeile bis zum Ende durch Leerraum gefüllt werden, geschieht das durch `\hspace{\fill}`. Eine Kurzform dieses Befehls ist `\hfill`.

Für vertikale Abstände gibt es entsprechend die Befehle `\vspace{Längenangabe}` bzw. `\vfill`.

Zeilen können nicht nur mit Leerraum gefüllt werden, sondern auch mit Linien und Punkten. Dafür gibt es die Befehle

`\hrulefill`..... `\hrulefill`
und `\dotfill`.

Mit diesen Befehlen kann man eine Zeile komfortabel in Bereiche teilen:

links.....das ist die Mitte.....rechts

9.2.5. Seiten

Ein Seitenwechsel wird durch den Befehl `\newpage` bewirkt.

TO DO !!!!!

Seitenzahlen

Änderung; Vorspann; Anhang

TO DO !!!!!

10. Die Gliederung eines Dokuments

10.1. Die Dokumentenklassen

10.1.1. Standardklassen

Im Standard- \LaTeX gibt es die Klassen `book`, `report`, `article` und `letter`.

Die Buch-Klasse wird für Bücher genommen. Klar! Die Buch-Klasse ist die einzige, die das „Kapitel“ `\chapter` als Gliederungsebene hat. Die Buchklasse ist dadurch ausgezeichnet, dass Kapitel (die in \LaTeX mit `\chapter{}` gesetzt werden) auf einer rechten Seite, also einer Seite mit einer ungeraden Seitenzahl, anfangen. Zur Not bleibt also auch einmal eine linke Seite frei.

Die weiteren Gliederungsebene sind der Abschnitt (`\section{}`), der Unterabschnitt (`\subsection{}`) und der Unterunterabschnitt (`\subsubsection{}`). In den geschweiften Klammern dieser Gliederungsbefehle werden die jeweiligen Überschriften angegeben, die automatisch ins Inhaltsverzeichnis übernommen werden.

Unterhalb des Unterunterabschnitts gibt es noch die Absätze und Unterabsätze (`\paragraph{}` bzw. `\subparagraph{}`). Die Inhalte der geschweiften Klammern werden nur noch hervorgehoben, aber nicht mehr numeriert und auch nicht mehr ins Inhaltsverzeichnis übernommen.

Die Report-Klasse unterscheidet sich von der Buch-Klasse dadurch, dass die Hierarchieebene des Kapitels nicht immer auf einer rechten Seite beginnt. Bei Buch und Report wird aber zwischen linker und rechter Seite unterschieden, zum Beispiel bei der Größe des rechten und des linken Randes. In der Regel ist der innere Seitenrand etwas kleiner als der äußere.

Die Artikel-Klasse besitzt nicht mehr das Kapitel als Gliederung. Hier wird auch nicht mehr zwischen geraden und ungeraden (linken und rechten) Seiten unterschieden.

10.1.2. Die Klassen von KOMA-Script

In der Erweiterung KOMA-Script, die eine Anpassung von \LaTeX an die Belange der deutschen Sprache zum Inhalt hat, gibt es zu jeder der allgemeinen Dokumentenklassen ein Pendant, zu `article` gibt es den `scartcl`, zu `report` den `scrreprt` und zu `book` den `scrbook`.

Diese Dokumentenklassen haben vor allem Auswirkungen auf die Gestaltung von Titelseiten (siehe Seite 90).

10.2. Gliederungsstufen und Überschriften

Grundsätzlich gilt, dass Überschriften in das Inhaltsverzeichnis aufgenommen werden. Aber wie bei vielen Dingen gibt es auch hier Ausnahmen. Wenn eine Überschrift eines Kapitels oder eines Abschnitts nicht ins Inhaltsverzeichnis aufgenommen werden soll, kann das ganz einfach mit Hilfe des „gesterten“ Befehls erreicht werden, d.h. statt zum Beispiel des Befehls `\section{blabla}` wird der Befehl `\section*{blabla}` benutzt.

Bei den ungesterten Befehlen wird in jeder Gliederungsstufe eine Ordnungsnummer mitgeführt und gesetzt. So bekommt die 4. „subsection“ in der 1. „section“ des 2. „chapters“ die Nummer 2.1.4. Bei den gesterten Gliederungsbefehlen entfällt die Numerierung. Selbstverständlich verwaltet \LaTeX die Zähler vernünftig, so dass bei einem neuen Eintrag in der nächsthöheren Gliederungsstufe der Zähler der nächstniedrigeren Stufe auf 0 zurückgesetzt wird.

Bei den Überschriften, die ins Inhaltsverzeichnis aufgenommen werden sollen, ist oft darauf zu achten, dass sie nicht länger als eine Zeile sind. Eintragungen im Inhaltsverzeichnis, die über 2 oder mehr Zeilen gehen, sehen sehr unschön aus. Zu diesem Zweck gibt es für die Gliederungsbefehle ohne Stern einen optionalen Parameter „Kurzform“. Ein Beispiel dazu:

`\section[Unsinn]{Eine viel zu langer Titel, der eigentlich unsinnig ist}` setzt die Überschrift des Abschnitts „Eine viel zu langer Titel, der eigentlich unsinnig ist“, während im Inhaltsverzeichnis einfach „Unsinn“ steht.

Wenn für das gesamte Dokument die Numerierung und die Aufnahme der Überschriften ins Inhaltsverzeichnis abweichend vom Standard geregelt sein soll, wird das im Vorspann erledigt. Dazu müssen zwei Zähler einen neuen Wert erhalten. Der Zähler `secnumdepth` gibt an, bis zu welcher Tiefe Überschriften numeriert werden sollen, der Zähler `tocdepth`, bis zu welcher Tiefe sie in das Inhaltsverzeichnis übernommen werden sollen. Die Befehle dazu lauten

`\setcounter{secnumdepth}{Zahl}` bzw. `\setcounter{tocdepth}{Zahl}`. Dabei gilt folgende Zuordnung für die Zahlen in der zweiten geschweiften Klammer:

chapter	0
section	1
subsection	2
subsubsection	3
paragraph	4
subparagraph	5

Tabelle 10.1.: Kennziffern für die Gliederungstiefe

Die Schranken sind standardmäßig für „book“ und „report“ auf 2 gesetzt, für „article“ auf 3.

10.2.1. Ein schlauer Spruch vorneweg

Don't ask me nothin' about
nothin'. I just might tell you the
truth.¹

(BOB DYLAN)

Manchmal möchte man an den Anfang eines Kapitels einen Hinweis geben auf das, was gleich passieren soll, und dazu fällt dem Autor ein, dass irgendeine wichtige Persönlichkeit zu diesem Thema schon mal etwas gesagt hat. Also los, das zitieren wir. Das sollte so gesetzt werden wie der Spruch zu Beginn dieses Kapitels — auch wenn diese Zeile aus einem Dylan-Song nichts mit diesem Kapitel zu tun hat. Die obigen Zeilen wurden so gesetzt:

```
1 \dictum [ \textsc{Bob Dylan} \\
2 { \textit{Outlaw Blues auf Subterranean Homesick Blues} } ]
3 {Don't ask me nothin' about nothin' .
4 I just might tell you the truth.}
```

Listing 10.1: Schlaue Sprüche

Eine leicht andere Darstellung bekommt man mit dem „epigraph“-Befehl:

```
1 \epigraph{Don't ask me nothin' about nothin' .
2 I just might tell you the truth.}
3 { \textit{Outlaw Blues auf Subterranean Homesick Blues} \\
4 \textsc{Bob Dylan} }
```

Listing 10.2: Schlaue Sprüche: Variante

10.3. Inhaltsverzeichnis

Über das Inhaltsverzeichnis eines Dokuments freut sich der \LaTeX -Schreiber besonders. Er hat nämlich im Normalfall überhaupt keine Arbeit damit. Wie oben schon gesagt, werden die Kapitelüberschriften automatisch ins Inhaltsverzeichnis übernommen, \LaTeX berechnet die Seiten und schreibt ins Inhaltsverzeichnis die richtige Seitenzahl. Allerdings muss man dabei beachten, dass \LaTeX erst nach Ende der ganzen Arbeit weiß, was auf welcher Seite steht. Das Inhaltsverzeichnis steht aber normalerweise am Beginn eines Dokuments. Damit die Seitenangaben stimmen, sollte man sicherheits halber also \LaTeX , das heißt das Programm `pdflatex` zweimal laufen lassen, damit die richtigen Seitenzahlen in das Inhaltsverzeichnis eingetragen werden.

Das Inhaltsverzeichnis wird also automatisch geschrieben — aber leider nicht automatisch gesetzt, bemerken viele \LaTeX -Anfänger. Deshalb muss zu Beginn des Dokuments der Befehl `\tableofcontents` abgesetzt werden.

¹in: Outlaw Blues auf Subterranean Homesick Blues

10. Die Gliederung eines Dokuments

Handarbeit ist nur dann erforderlich, wenn in das Inhaltsverzeichnis mehr oder auch weniger aufgenommen werden soll, als es im Standard vorgesehen ist. Wie oben geschrieben, ist das „Weniger“ leicht zu erreichen: die gesternten Gliederungsbefehle veranlassen L^AT_EX, die Kapitelüberschriften nicht ins Inhaltsverzeichnis zu übernehmen.

Mehr Eintragungen ins Inhaltsverzeichnis erreicht man durch den Befehl `\addtocontents{toc}{Das ist eine Eintragung}`

TO DO !!!!!

10.4. Über Überschriften

Zeilenumbrüche in Überschriften: muss das sein?!? Überschriften sollen eigentlich prägnant sein und kein Romanformat haben! Manchmal lässt es sich aber nicht vermeiden, dass eine Kapitelüberschrift über mehrere Zeilen geht. Dann ist es aber oft nötig, dass der Autor den Zeilenumbruch von Hand macht, damit die Überschrift aussagekräftig ist. Zeilenumbrüche in Überschriften bringen aber L^AT_EX ins Schleudern. Deshalb müssen Zeilenumbrüche in Überschriften mit Hilfe des Befehls `\protect` geschützt werden. Der Befehl `\protect` wird direkt vor dem zu schützenden Befehl eingefügt. Man schreibt also

```
\section{Eine eigentlich viel zu lange Überschrift\protect\\für ein Kapitel}
```

10.5. Sonstiges

10.5.1. Anhang

Ein Anhang eines Dokuments ist etwas, was nicht zur eigentlichen Information gehört, aber diese vielleicht ergänzt, erläutert, oder, wie im Fall dieses Dokuments, die rechtliche Seite des Schreibens erläutert.

Leere Seite: `\clearpage`,

mit Gewalt: `~\clearpage`

Schusterjungen: `\clubpenalty=10000` Hurenkinder: `\widowpenalty=10000`

Zu viele floats auf einer Seite: `\setcounter{topnumber}{4}` erlaubt 4 floats am Seitenkopf ebenso `\setcounter{bottomnumber}{4}` und `\setcounter{totalnumber}{4}` und `\renewcommand{\topfraction}{.9}` setzt maximalen Anteil floats am Seitenkopf

TO DO !!!!!

TO DO !!!!!

TO DO !!!!!

11. Der Kleinkram vorne und hinten dran

11.1. Abbildungsverzeichnis

Für ein Abbildungsverzeichnis ist nichts zu machen! Alles, was in einer `picture`-Umgebung steht, kommt in ein Abbildungsverzeichnis. Das Abbildungsverzeichnis muss nur noch gesetzt werden, was durch den Befehl `\listoffigures` geschieht.

11.2. Tabellenverzeichnis

Und für das Tabellenverzeichnis gilt das selbe wie für das Abbildungsverzeichnis: alles innerhalb einer `tabular`-Umgebung kommt hinein, gesetzt wird es durch `\listoftables`.

11.3. Index (Stichwortverzeichnis)

Stichwortverzeichnisse sind vor allem in Lehrbüchern fast unerlässlich. \LaTeX unterstützt auch hierbei den Schreiber vortrefflich, weil außer dem Eintrag selber fast nichts gemacht werden muss. Das Zusatzprogramm `\makeindex` verwaltet alle Indexeinträge, sortiert sie alphabetisch und verwaltet die Seiten, auf die die Indexeinträge verweisen. Allerdings sind Indexeinträge dadurch problematisch, dass durch eine Neugenerierung eines Dokuments die Verweise auf eine falsche Seite zeigen.

11.3.1. Die Einträge

Alle Beispiele dieses Abschnitts werden auch als Indexeinträge für dieses Buch durchgeführt. Die „Elefanten“ tauchen also in dem Stichwortverzeichnis für \LaTeX auf, auch wenn sie so gar nichts mit \LaTeX zu tun haben. Aber so kannst Du, der Leser, sofort sehen, wie ein solcher Eintrag aussieht.

Einträge in das Stichwortverzeichnis können auf drei verschiedene Arten gemacht werden. Ein einfacher Eintrag geschieht mit dem Befehl `\index{Elefant}` Damit wird der Eintrag „Elefant“ in das Stichwortverzeichnis gemacht.

Soll ein untergliederter Eintrag gemacht werden, so geschieht das durch `\index{Elefant!indischer}`. Das Ergebnis ist der Haupteintrag „Elefant“, in der Zeile darunter der Unter-Eintrag „indischer“ mit der dazugehörigen Seitenzahl.

11. Der Kleinkram vorne und hinten dran

Soll ein Begriff zwar im Stichwortverzeichnis auftauchen, dort aber auf einen anderen Begriff verwiesen werden, geschieht das durch `\index{afrikanischer Elefant|see{Elefant}}`.

11.3.2. Die Erzeugung des Stichwortverzeichnisses

Das Stichwortverzeichnis wird in mehreren Schritten erzeugt. In der Hauptdatei wird der Befehl `\makeindex` zu Beginn des Dokuments eingegeben. Durch einen \LaTeX -Lauf wird die Hauptdatei generiert und eine Datei mit der Endung „.idx“ angelegt, in die alle Indexeinträge mit zugehöriger Seitenzahl geschrieben werden. Der Dateiname selber ist identisch mit dem Dateinamen der Hauptdatei.

Danach wird das Zusatzprogramm `makeindex` aufgerufen, das aus der „.idx“-Datei eine „.ind“-Datei erstellt, in der das fertige Stichwortverzeichnis mit den dazugehörigen \LaTeX -Befehlen steht.

In die Hauptdatei wird an die Stelle, wo das Stichwortverzeichnis erscheinen soll, also meistens an das Ende des Dokuments der Befehl `\printindex` eingefügt.

Danach muss noch mindestens zwei Mal der \LaTeX -Lauf durchgeführt werden, damit die Referenzen auf die korrekten Seiten aufgelöst werden.

11.4. Glossar (Schlagwortverzeichnis)

11.4.1. Die Einträge

Für das Erstellen eines Schlagwortverzeichnis gibt es unter L^AT_EX auch einige Pakete. Hier beschreibe ich das Paket `glossaries`, das viele Möglichkeiten, nicht nur für ein Glossar, bietet. Mit Hilfe dieses Pakets können auch Abkürzungsverzeichnisse erstellt werden. Deswegen wird im weiteren Text von „den Stichwortverzeichnissen“ gesprochen, um alle diese Verzeichnisse einzuschließen.

Das Vorgehen gliedert sich wieder in mehrere Schritte. Ich empfehle, die Begriffe, die in das Stichwortverzeichnis aufgenommen werden sollen, in einer eigenen Datei zu speichern. Einträge in das Glossar entstehen durch den Befehl `\newglossaryentry{}`. Dieser Befehl hat 2 Parameter. Der erste Parameter gibt den Markierungsnamen an, unter dem dieser Eintrag zu finden sein soll. Es empfiehlt sich hier, Glossareinträge mit einem Präfix, hier „gl“ zu versehen. Der zweite Parameter besteht aus einer Liste von Schlüssel-Wert-Paaren. Verpflichtend für ein Glossar sind natürlich die Schlüssel „name“ und „description“. Die Werte werden durch ein Gleichheitszeichen direkt an den Schlüssel angehängt. Wie unten ersichtlich, wird ein Wert, der aus mehreren Wörtern besteht, in geschweifte Klammern geschrieben.

Die Einträge in einer solchen Datei sehen so aus:

```

1 \newglossaryentry{glKnuth}{name=Knuth ,
2   description={Donald E. Knuth ,
3   amerikanischer Mathematiker ,
4   geboren 1938 , wurde weniger durch
5   seine mathematischen Leistungen
6   sondern mehr durch seine Arbeiten
7   auf dem Gebiet der Informatik bekannt.}
8 }
```

Listing 11.1: Glossar-Eintrag

Soll ein solcher Eintrag mehrere Absätze umfassen, ist der normale Zeilenvorschub durch `\\` möglich.

Auch Abkürzungen, hier Akronyme genannt, stehen sinnvollerweise in einer eigenen Datei. Die Einträge werden so geschrieben:

```

1 \newacronym{CD}{CD}{Compact Disc }
```

Listing 11.2: Eintrag ins Abkürzungsverzeichnis

Dabei ist der erste Parameter der Markierungsname, der zweite die Eintragung in das Abkürzungsverzeichnis und der dritte die Bedeutung der Abkürzung.

11.4.2. So wird es eingebunden

Im Hauptdokument muss zuerst das Paket `glossaries` eingebunden werden. Das geschieht mit dem Befehl `\usepackage{glossaries}`. Dabei ist darauf zu achten, dass

11. Der Kleinkram vorne und hinten dran

dieses Paket erst nach dem Paket `hyperref` eingebunden wird.

Wenn man zusätzliche Möglichkeiten des Paketes „`glossaries`“ benutzen will, werden diese Möglichkeiten durch optionale Parameter bereitgestellt. Die Stichwortverzeichnisse in diesem Buch wurden durch

```
1 \usepackage[nonumberlist, acronym, toc, section]{glossaries}
```

Listing 11.3: Einbinden der Glossar-Umgebung

erstellt. Die Bedeutung der einzelnen Parameter ist

nonumberlist es werden keine Seitenzahlen der Seiten, auf denen der Begriff auftaucht, eingefügt

acronym ein Abkürzungsverzeichnis wird erstellt

toc die Stichwortverzeichnisse werden ins Inhaltsverzeichnis aufgenommen

section auf der Section-Ebene

Die Erstellung der Stichwortverzeichnisse erfolgt mit dem Befehl `\makeglossaries`.

Erst nach diesem Aufruf bindet man die externen Dateien ein mit

```
1 \loadglsentries{glossar}  
2 \loadglsentries{akronyme}
```

Listing 11.4: Import von Glossar und Abkürzungsverzeichnis

11.4.3. Generieren der Schlagwortverzeichnisse

Das Paket `glossaries` enthält ein perl-Skript, das alle möglichen Stichwortverzeichnisse generiert. Es heißt `makeglossaries` und wird mit der Hauptdatei des Dokuments als Parameter aufgerufen, also etwa `makeglossaries DasBuch`

Als letzter Schritt fehlt noch das Setzen der Stichwortverzeichnisse. Diese Verzeichnisse stehen gewöhnlich am Ende eines Dokumentes. Dazu dienen die Befehle `\printglossary[style=altlist,title=Glossar]` für das Glossar und

```
1 \renewcommand*{\acronymname}{Abkürzungsverzeichnis}  
2 \deftranslation[to=German]{Acronyms}{Abkürzungsverzeichnis}  
3 \printglossary[type=\acronymtype,style=long]
```

Listing 11.5: Drucken des Abkürzungsverzeichnisses

für das Abkürzungsverzeichnis.

11.5. Literaturverzeichnis

Auch für das Literaturverzeichnis gibt es wieder mehrere Möglichkeiten:

1. die handgestrickte Variante
2. das mächtige Bib \TeX
3. das mächtige und komfortable `biblatex`

Die handgestrickte Variante für ein Literaturverzeichnis benutzt die `thebibliography`-Umgebung und ansonsten einige \LaTeX - Befehle. Diese Variante ist zu vergleichen mit einer Datei der benutzten Literatur. Bib \TeX ist ein Zusatzprogramm zu \TeX , dessen Ergebnis zu vergleichen ist mit einer Datenbank. `biblatex` ist eine Hülle um Bib \TeX , die die Benutzung von Bib \TeX ähnlich vereinfacht, wie \LaTeX die Benutzung von \TeX vereinfacht hat.

11.5.1. Eigene Literaturverzeichnisse

Das handgestrickte Literaturverzeichnis ist eine Liste von Literatur-Einträgen in einem beliebigen \LaTeX - Dokument. Die Liste wird in die Umgebung `thebibliography` eingebunden. Als Parameter wird dem `\begin{thebibliography}` eine ganze Zahl mitgegeben, deren Stellenanzahl die maximale Anzahl der Einträge in diese Liste festlegt. Eine übliche Deklarartion ist zum Beispiel

```

1 \begin{thebibliography}{999}
2
3 \end{thebibliography}

```

Listing 11.6: Umgebung für bibliography

Die `items` heißen hier `bibitem`. Jedes `bibitem` bekommt in eckigen Klammern eine Ordnungsnummer, die dann bei einer Zitierung des Literatureintrages in den Text eingefügt wird. Es folgt eine Kurzbezeichnung des Literatureintrages. Die einzelnen Attribute eines `bibitem`-Eintrags werden dann nach dem Geschmack des Schreibers formatiert.

11. Der Kleinkram vorne und hinten dran

Hier kommt ein Vorschlag, wie man so etwas machen könnte:

```
1 \begin{thebibliography}{999}
2   \bibitem[1]{Kapital} {\sc Marx, Karl}: {\em Das Kapital},
3     {Verlag der Republik, 1958}
4   \bibitem[2]{Schnaps} {\sc May, Karl}:
5     {\em Der Schnaps im Silbersee},
6     {alco-Verlag, 1989}
7   \bibitem[3]{Kapitol} {\sc Marx, Groucho}: {\em Das Kapitol},
8     {Monty-Python-Verlag, 2008}
9   \bibitem[4]{Kapitaelchen} {\sc Gensfleisch, Johannes}:
10     {\em Die Kapitäelchen als Zierschrift},
11     {Helau-Verlag, 1456}
12 \end{thebibliography}
```

Listing 11.7: Handgestrickte Literaturangaben

Hier wird der Name des Autors in Kapitälchen gesetzt, der Titel des Werks mit Hilfe von `\em` in Schrägschrift, der Rest in normaler Schrift. Die Verweise im laufenden Text sind in diesem Fall die natürlichen Zahlen in eckigen Klammern. Falls man diese Eintragungen ändert, zum Beispiel auf einen sprechenden Verweis wie „Marx58“ für das erste Werk, ist es in der Verantwortung des Autors, dass diese Schlüssel eindeutig sind.

Einen Literaturhinweis macht man im laufenden Text durch den Befehl `\cite{Ordnungsnummer}`.

Für kurze Literaturverzeichnisse ist das durchaus angemessen, und die Darstellung kann beliebig gewählt werden. Allerdings ist damit die Intelligenz von $\text{T}_{\text{E}}\text{X}$ / $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ nicht ausgeschöpft; eine Sortierung der Einträge im Literaturverzeichnis muss manuell gemacht werden, die Gefahr eines doppelten Schlüssels ist gegeben und viele andere Fallen lauern.

11.5.2. Bib $\text{T}_{\text{E}}\text{X}$

Besser ist es, $\text{T}_{\text{E}}\text{X}$ / $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ beziehungsweise das Zusatzprogramm Bib $\text{T}_{\text{E}}\text{X}$ seine Arbeit machen zu lassen. Große Vorteile davon sind, dass

- das Literaturverzeichnis in einer eigenen Datei abgelegt wird
- man dadurch parallel an seinem Schriftstück und der Literaturliste arbeiten kann; das Zusammenführen von zitierter Literatur und dem eigenen Werk geschieht bei der Generierung des Dokumentes
- die Einträge darin einer vorgegebenen Form entsprechen
- Sortierung vom Programm erledigt wird
- es unzählige verschiedene Stile für die Darstellung des Literaturverzeichnisses und der Zitierung im Text gibt

- ein Literaturverzeichnis für viele verschiedene Werke benutzt werden kann; Bib_TE_X nimmt in das Literaturverzeichnis eines Werkes nur die Einträge aus der separaten Datei auf, die tatsächlich zitiert werden, aber man hat als Schreiber die Möglichkeit, weitere Werke einzubinden

Da wie oben erwähnt, das Literaturverzeichnis in einer eigenen Datei gespeichert wird, soll hier zuerst dieser Schritt erklärt werden. Die Datei, die das Literaturverzeichnis enthält, muss als Endung `bib` haben. Ein möglicher Dateiname ist also `meineLit.bib`.

Für die Einträge gibt es vorgeschriebene Dokumentenarten. Jeder Eintrag in das Literaturverzeichnis beginnt mit der Festlegung der Art des Eintrages, dem ein Klammersaffe `@` vorangestellt wird. Die wichtigsten Arten von Dokumenten sind

- ARTICLE
- BOOK
- INBOOK (für einen Artikel, ein Kapitel oder Seiten eines Buches)
- MASTERTHESIS (für eine Diplomarbeit, Doktorarbeit o.ä.)
- UNPUBLISHED (für nicht veröffentlichte Werke)

Weitere Dokumentenarten kann man im Internet nachschlagen, wenn eine der obigen den Ansprüchen nicht genügt.

Es gibt zu jeder Dokumentenart vorgeschriebene Attribute. Ich erwähne hier nur die wichtigsten Attribute, die bei jeder der vorher genannten Dokumentenarten zulässig sind. Es sind diese

- AUTHOR
- TITLE
- YEAR

Für die einzelnen Dokumentenarten gibt es dann noch spezielle Attribute, die sich logisch ergeben: Artikel in wissenschaftlichen Zeitschriften haben das Attribut `JOURNAL`, Diplomarbeiten u.ä. das Attribut `SCHOOL` usw.

11. Der Kleinkram vorne und hinten dran

Hier folgt ein Beispiel, wie die Literaturliste in einer `bib`-Datei aussehen könnte:

```
1 @book{Goossens98,  
2   title = {Der LATEX-Begleiter},  
3   author = {Goossens, Michael  
4             and Mittelbach, Frank  
5             and Samarin, Alexander},  
6   publisher = {Addison Wesley Verlag},  
7   isbn = {3-89319-646-3},  
8   pages = 554,  
9   keywords = {LaTeX}  
10 }  
11  
12 @book{kopka-liemeeim1995,  
13   title = {LATEX II. Ergänzungen,  
14           mit einer Einführung in METAFONT},  
15   author = {Kopka, Helmut},  
16   publisher = {Addison Wesley Verlag},  
17   isbn = {3-89319-665-X},  
18   pages = 428,  
19   keywords = {LaTeX}  
20 }
```

Listing 11.8: Beispiel für Literatureinträge

Ein Eintrag beginnt also mit der Festlegung des Typs, das zitiert wird. Es folgt ein eindeutiger Schlüssel, unter dem beim Zitieren das Werk angesprochen wird. Danach kommen eine Reihe von Feldern, die für den jeweiligen Dokumententyp zulässig sind. Dabei sind die beiden Schreibweisen `attributname = {wert}` und `attributname='wert'` erlaubt. Jedes Attributname-Wert-Paar wird vom folgenden durch ein Komma getrennt.

Auch bei dieser Form des Literaturverzeichnisses macht man einen Literaturhinweis im laufenden Text durch den Befehl `\cite{Schlüssel}`, wie man sieht aber hier mit dem Schlüssel des Eintrags als Parameter.

Ein großer Vorteil der Erstellung von Literaturverzeichnissen mit BibTeX besteht darin, dass viele, auch einfache „Datenbanken“ für die Verwaltung persönlicher Sammlungen die Möglichkeit eines Exportes nach BibTeX bieten. Wer also sowieso seine ganze Haus-Bibliothek mittels eines solchen Programms verwaltet, spart sich die Tipparbeit für Einträge in ein solches Verzeichnis. Unter Linux leisten dies zum Beispiel die Programme `tellico` und `GCStar`.

Die Stile für das Literaturverzeichnis und die Zitierweise stehen in `.bst`-Dateien, wobei das `.bst` für „bibliography style“ steht. Diese Dateien stehen unter Unix-Betriebssystemen im Verzeichnis `/usr/share/texmf/bibtex/bst` und seinen Unterverzeichnissen. Oft benutzte Stile sind `plain` und `alpha`.

Das Einbinden und die Generierung des Literaturverzeichnisses, das mittels BibTeX erstellt und aus dessen Werken zitiert wird, ist nicht so einfach wie bei der handgestrick-

ten Methode. Das Literaturverzeichnis steht in der Regel am Ende eines Dokuments, und an der Stelle, an der es eingefügt werden soll, müssen die Befehle

```
1 \bibliography{dateiname}
2 \bibliographystyle{stil}
```

Listing 11.9: Befehle für die Einbindung des Literaturverzeichnisses

stehen. Der Dateiname ist der oben erwähnte `.bib`-Name, als Stil wählt man, sofern man nicht eine spezielle Vorgabe bekommt, wahrscheinlich `alpha` oder `plain`.

Gerade der Aspekt des Zitierstils ist bei wissenschaftlichen Arbeiten nicht zu vernachlässigen. Wenn man sein Werk in einer Zeitschrift seines Fachbereiches veröffentlichen will, wird einem der zu wählende Stil vorgegeben, so dass bei den Artikeln der Fachzeitschrift Einheitlichkeit herrscht. Aber auch bei Diplom-, Bachelor-, Masterarbeiten und ähnlichem ist es sehr zu empfehlen, sich vorher zu erkundigen, welche Art der Zitierung und welche Form der Einträge im Literaturverzeichnis gewünscht wird. Akademiker sind manchmal ganz schön heikel!

Die Generierung des Literaturverzeichnisses geschieht in vier Schritten, weil hier wieder Querverbindungen hergestellt werden müssen. So geht es:

1. `pdflatex dateiname`
2. `bibtex dateiname`
3. `pdflatex dateiname`
4. `pdflatex dateiname`

Erst nach dem dritten `pdflatex`-Lauf stimmen alle Seitenzahlen!

11.5.3. bib \LaTeX

`bib \LaTeX` bedeutet nochmals eine Vereinfachung, da hier viel Handarbeit, die man bei `Bib \TeX` noch leisten musste, hinter einer \LaTeX -ähnlichen Syntax versteckt wird.

Zuerst muss man die Pakete einbinden, die `bib \LaTeX` enthalten und die, die `bib \LaTeX` benötigt. Das geschieht durch

```
1 \usepackage[style=authortitle-icomp]{biblatex}
2 \usepackage[babel,german=guillemets]{csquotes}
3 \bibliography{/home/fmartin/texte/Buecher/meineBuecher.bib}
```

Listing 11.10: Befehle im Vorspann für `biblatex`

Für die Erstellung des Literaturverzeichnisses gilt alles, was oben bei `Bib \TeX` gesagt wurde. Insbesondere die Schreibweise der Eintragungen in das Literaturverzeichnis sind identisch. Allerdings gibt es weitere Dokumententypen, von denen hier nur `@online` erwähnt wird. Dieser Dokumenttyp benötigt die Attribute `author`, `title`, `year` und `url`.

11. Der Kleinkram vorne und hinten dran

Vor allem der Aufrufmechanismus vereinfacht sich: es muss nur noch an der Stelle, an der das Literaturverzeichnis angezeigt werden soll, also meistens am Ende des Dokuments, durch `\printbibliography` das Literaturverzeichnis eingebunden werden. Die (mehrfachen) Aufrufe der verschiedenen L^AT_EX-Programme werden hier durch das bibL^AT_EX-Paket übernommen.

Das Handbuch zu bibL^AT_EX findet man im Internet¹. Auch für die Anpassung der bibL^AT_EX-Styles verweise ich auf das Internet.²

11.5.4. Besonderheiten

Einfügen der nicht zitierten Werke durch `\nocite{*}`. Das muss im Falle der Benutzung von BibT_EX vor dem Befehl `\bibliography{meineBib}` stehen, im Falle, dass man bibL^AT_EX benutzt, vor dem `\printbibliography`.

¹`biblatexpack`.

²`biblatexstyle`.

12. Was sonst noch so alles geht mit \LaTeX

12.1. Präsentationen . . . die immer funktionieren

Präsentationen? Das Wort zu erwähnen ist fast so schlimm wie das Wort Textverarbeitung auszusprechen. In beiden Fällen taucht fast unweigerlich eine Schleichwerbung für ein Produkt einer großen Firma der Softwarebranche aus dem Westen der USA auf.

Über den Sinn der Benutzung eines Programms zur Erstellung einer Bildschirm-Präsentation sollte man wirklich nachdenken. Dabei helfen kann das Buch von Coy¹. Grundsätzlich gilt, dass eine Bildschirm-Präsentation nur eine Ergänzung des Gesagten sein sollte, den Text weder paraphrasieren noch ersetzen sollte. Der visuelle Impuls sollte dabei nicht das Gesagte überdecken oder davon ablenken. Kurz gesagt: auf den Inhalt kommt es an.

Immer wieder Freude bereitet es², wenn ein Referent ins Haus kommt, seinen eigenen Rechner nicht dabei hat, und dann feststellen muss, dass die zu Hause mühselig erstellte Präsentation auf dem vorhandenen Rechner nicht so funktioniert wie sie sollte³.

Dabei geht es auch anders. Präsentationen, die mit \LaTeX erstellt werden, benötigen nur einen pdf-Betrachter. Wer dann auch noch Präsentationen immer mal wieder ändern muss, wird sich auch freuen, wenn er eine einfache, gut überschaubare Textdatei vor sich hat. Oft reicht schon eine Veränderung im Vorspann der Präsentation, um eine Änderung vorzunehmen. Also: rein ins Vergnügen!

Ein ausführliches Handbuch findet man unter einem Knoten des Comprehensive TeX Archive Network (CTAN), zum Beispiel dem in⁴ Großbritannien.

¹Coy09.

²weil man in der Zwischenzeit Kaffee trinken kann!

³bestenfalls!! Manchmal funktioniert sie überhaupt nicht.

⁴beameruser.

12.1.1. Die Klasse Beamer

Für Präsentationen gibt es eine eigene Klasse, die Klasse `beamer`. Die erste Zeile des Vorspanns einer Präsentation sieht also so aus:

`\documentclass[10pt,german,ngerman,utf8,a4paper]{beamer}` Wie bei allen anderen \LaTeX -Dokumenten werden dann Pakete eingebunden, neu ist hier, dass mit

```
1 \usetheme[hideothersubsections]{Marburg}
2 \usecolortheme{whale}
3 \usefonttheme{structurebold}
4 \useinnertheme{rounded}
```

Listing 12.1: Vorspann von `beamer`

spezielle Befehle für die Darstellung im Vorspann stehen.

Jede Präsentation hat ein bestimmtes Aussehen, ein „theme“. In \LaTeX sind die themes nach (Universitäts-)Städten benannt.⁵ Es gibt

- AnnArbor
- Antibes
- Bergen
- Berkeley
- Berlin
- Boadilla
- CambridgeUS
- Copenhagen
- Darmstadt
- Dresden
- Frankfurt
- Freiburg
- Goettingen
- Hannover
- Ilmenau
- JuanLesPins
- Luebeck
- Madrid
- Malmoe
- Marburg
- Montpellier
- PaloAlto
- Pittsburgh
- Rochester
- Singapore
- Szeged
- Warsaw

Das theme wird mit dem Befehl `\usetheme` benützt.

Besonders hervorzuheben ist der optionale Parameter `hideothersubsections`, der bewirkt, dass die Unterkapitel im Inhaltsverzeichnis ausgeblendet werden außer der Unterkapitel des aktuellen Kapitels

Für viele dieser themes gibt es dann noch ein Farbschema, ein `colortheme`. Die Farbschemata sind nach Tieren benannt:

- albatross
- beaver
- beetle
- crane
- dove
- fly
- seagull
- wolverine

⁵Tübingen fehlt!!

Auch die Standardschrift wird im Vorspann eingestellt, und zwar durch den Befehl `\usefonttheme{schrift}`. An Schriften stehen zur Verfügung:

- professionalfonts
- serif
- structurebold
- structureitalicserif
- structuresmallcapserif

Ferner gibt es noch „innerthemes“, die

- default
- circles
- rectangels
- rounded
- inmargin

`innertheme` bestimmt, wie zum Beispiel Aufzählungen, Überschriften und anderes dargestellt werden.

12.1.2. Die „Titelseite“

Eine Folie wird in die Umgebung `frame` eingebettet. Auch in Präsentationen ist es gut, wenn man eine Angabe über Autor und Thema macht! Die Titelseite wird hier genauso gesetzt wie in anderen Dokumenten. Dazu benutzt man wieder die Befehle

```

1  \author{Mein Name}
2  \title{So geht es}
3  \date{\today}
4  \begin{frame}
5     \titlepage
6  \end{frame}

```

Listing 12.2: Titelseite in beamer

Anders als in einem normalen Dokument wird daraus aber keine echte Titelseite gesetzt, sondern auf jeder Folie je nach „theme“ Name und Titel angegeben.

12.1.3. Das Inhaltsverzeichnis

Das Inhaltsverzeichnis bei einer Präsentation dient dazu, den Zuhörern schon vorweg zu erklären, was man mit der Präsentation erreichen will und welche Gesichtspunkte man anschneiden wird. Das setzen des Inhaltsverzeichnisses ist aber keine Neuigkeit:

```

1  \begin{frame}{Inhalt}
2     \tableofcontents
3  \end{frame}

```

Listing 12.3: Inhaltsverzeichnis in beamer

12.1.4. Die einzelnen Folien

Als Beispiel wird hier eine einzelne Folie gezeigt:

```
1 \section{Grüß Gott}
2 \begin{frame}{Grüß Gott}
3 \begin{itemize}
4 \item<1-> .. und los gehts mit
5 \item<2-> auf, weiter
6 \end{itemize}
7 \end{frame}
```

Listing 12.4: Eine Folie

Diese Folie beginnt ein neues Kapitel. Für das Inhaltsverzeichnis der Präsentation gelten ähnliche Aussagen wie bei Inhaltsverzeichnissen anderer Dokumente. Der Unterschied ist, dass bei Folien das Inhaltsverzeichnis keine extra Seite einnimmt, sondern je nach Ansicht auf jeder Folie zu sehen ist, oft im Randstreifen.

Innerhalb der Folie, und das ist ja das typische für Präsentationen, werden Aufzählungen geschrieben, deren einzelne Punkte nacheinander einblendend werden. Das sind also die klassischen Aufzählungslisten oder nummerierte Listen. Der Unterschied ist, dass der Befehl `\item` leicht verändert aussieht: hinter dem `item` steht in spitzer Klammer eine Zahl, gefolgt von einem Bindestrich, also `\item<1->`. In der Reihenfolge dieser Zahlen werden die einzelnen Punkte einblendend, in der Regel durch Drücken der Enter-Taste oder einer Maus-Taste.

Eine solidere Methode, die noch dazu mit weniger Schreibarbeit verbunden ist, setzt das Einblenden im `itemize`-Kopf. Das sieht dann so aus:

```
1 \section{Grüß Gott}
2 \begin{frame}{Grüß Gott}
3 \begin{itemize}[<+>]
4 \item .. und los gehts mit
5 \item auf, weiter
6 \end{itemize}
7 \end{frame}
```

Listing 12.5: Einfacheres Einblenden

12.1. Präsentationen ... die immer funktionieren

Um den jeweils aktuellen Punkt der Präsentation besonders hervorzuheben, kann man einen weiteren Parameter an die Listen-Umgebung (`itemize` oder `enumerate`) übergeben.

```
1 \section{Grüß Gott}
2 \begin{frame}{Grüß Gott}
3 \begin{itemize}[<+ -| alert@+>]
4 \item .. und los gehts mit
5 \item auf, weiter
6 \end{itemize}
7 \end{frame}
```

Listing 12.6: Hervorhebung des aktuellen Punktes

Mit Hilfe des Parameters `alert@+` wird der aktuelle Punkt der Präsentation immer in roter Farbe angezeigt.

12. Was sonst noch so alles geht mit \LaTeX

12.2. Musik, zwei, drei

12.2.1. Lilypond

The image shows two musical staves with guitar tablature. The first staff is in treble clef, common time (C), and contains a melody with a triplet of eighth notes and a quarter note. The guitar part below it shows fret numbers: 12 on the bass string, 12 on the A string, and 9-10-12-12-14-14-12-13 on the treble strings. A triplet bracket is shown over the first three notes of the treble staff. The second staff is also in treble clef, common time, and contains a melody with eighth notes and a final chord. The guitar part shows fret numbers: 3-0-2-3 on the bass string, 8-5-7-8-5 on the treble strings, and a final chord of 3-5-5. A small '8' is written below the first note of the melody in the second staff.

Das sieht doch wirklich klasse aus! Auch Noten können mit \LaTeX überzeugend gesetzt werden. Das ist mit reinem \TeX / \LaTeX nicht ganz einfach, obwohl es — wie alles bei \TeX — logisch ist.

Zum Glück gibt es ein Zusatzpaket zu \LaTeX , das die Arbeit einfach macht: Lilypond, initiiert und geschrieben von den zwei niederländischen Musikern, Programmierern und \LaTeX -Liebhabern Han-Wen Nienhuys und Jan Nieuwenhuizen.

Das obige Beispiel sieht als Quelltext so aus:

```

1 \new StaffGroup <<
2 \new Staff \relative c'' {
3   \partial 4
4   a\5
5   \times 2/3 {d8\4 e\3 f\3} g4\3 \times 2/3{g16\3 r a\3} a4.\3
6   g4\3 gis\3 gis2
7 }
8
9 \new TabStaff \relative c' {
10  \time 4/4
11  \partial 4
12  a4\5
13  \times 2/3 {d8\4 e\3 f\3} g4\3 \times 2/3{g16\3 r a\3} a4.\3
14  g4\3 gis\3 gis2
15 }
16 >>
17
18 \new TabStaff \relative c {
19  <c\5 e\4 g\3 >
20  fis < e, a d g b e > % e, a, d
21 }
22
23 \new StaffGroup <<
24 \new Staff \relative c {
25   \clef "treble.8"
26   c16 d e f g4
27   c,16 d e f g4
28   f <<c'4 e g>>
29 }
30 \new TabStaff \relative c {
31   c16 d e f g4
32   \set TabStaff.minimumFret = #5
33   c,16 d e f g4
34   f <<c'4\3 e\2 g\1>>
35 }
36 >>

```

Listing 12.7: Notensatz mit Lilypond

Das hier soll keine Einführung in Lilypond werden. Aber die wichtigsten Gestaltungselemente werden erklärt.

Der erste Block besteht aus 2 Teilen, einer Notenzeile und einer Tabulaturzeile. Eine neue Notenzeile (siehe Zeile 2) ist ein `\new Staff`, eine neue Tabulaturzeile (siehe Zeile 9) ist ein `\new TabStaff`. Diese beiden Teile werden durch ein `\new StaffGroup`

12. Was sonst noch so alles geht mit \LaTeX

zusammengefasst.

In allen Zeilen im obigen Beispiel wird ein Bezugspunkt gesetzt. Das geschieht hier durch den Befehl `\relative c''`, wodurch der Bezugspunkt das zweigestrichene `c` ist. Die gesamte Notenzeile (bzw. die Tabulaturzeile), die sich auch über mehrere Zeilen hinziehen kann, wird in geschweifte Klammern eingefasst.

Das erste Beispiel fängt mit einem Auftakt an. Der Befehl dazu ist in Zeile 3 das `\partial 4`. Dies bedeutet einen Auftakt von einer Viertelnote. Die Viertelnote ist ein „a“, das auf der Gitarre auf der 5. Saite gespielt werden soll (die höchste Saite ist die 1. Saite, die tiefste die 6.), was in Lilypond durch eine hinter einem Backslash stehende „5“ angewiesen wird. Die Tabulatur für Gitarre setzt das dann entsprechend mit der Bundangabe „12“.

Für viele Eigenschaften des Notensatzes gibt es Standardvorgaben. So wird in diesem Beispiel keine Taktangabe gemacht. In diesem Fall wird einfach ein 4/4-Takt gesetzt. Ebenso gilt als Vorgabe für die Notenlänge eine Viertelnote.

Jeder Takt wird in eine separate Zeile geschrieben. In Zeile 5 fängt also der zweite Takt (der erste richtige nach dem Auftakt) an. Hier steht die nächste beachtenswerte Konstruktion: `\times 2/3 d84 e3 f3. \times 2/3` gibt an, dass die nächsten 3 Noten, die in der darauffolgenden geschweiften Klammer stehen, als Triole mit einem Längenwert von $\frac{2}{3}$ gespielt werden sollen. Die erste Note bekommt als Länge eine „8“ angehängt, was bedeutet, dass es sich um eine Achtelnote handelt. Die Längenangabe gilt ab jetzt so lange, bis eine andere Längenangabe ausdrücklich gemacht wird. Also sind auch die nächsten beiden Noten in der geschweiften Klammer Achtelnoten, und diese 3 Achtelnoten sind als Triolen zu spielen, haben also zusammen die Länge von 2 Achtelnoten oder einer Viertelnote.

12.3. Deutsche Schriften

Wie schon oben beim Thema Ligaturen erwähnt (s. Seite 41) stehen in L^AT_EX auch altdeutsche Schriften wie Gotisch, Fraktur, Schwabacher, Sütterlin . . . zur Verfügung. Diese werden mit dem Paket `oldgerman` eingebunden.

Unter Umständen gibt es dann Fehlermeldungen der Form ‚can’t find file ‘yfrak‘. Das hat zur Ursache, dass die Fonts dieser Schriften noch nicht erstellt wurden. Das wird auf der Shell mit Hilfe des Befehls `mktexmf ygoth12` (hier für die Gotik in Schriftgröße 12 Punkt) gemacht.⁶

Hier folgen einige Zeilen Text, die in Gotik gesetzt sind.

Alte deutsche Schrift wird heute fast nur noch als Zielschrift gelehrt. Die Geschichte der deutschen Schriften in der jüngsten Vergangenheit ist schon eigenartig. Wie man vermuten könnte, war in der schlimmsten Zeit der deutschen Geschichte, im Nationalsozialismus, alles Deutsche besonders gefördert, also auch Schriften wie Gotik oder Sütterlin. Und dann waren es viele besonders deutsche Deutschen, die per Geleß diese Schriften abschafften.

Schon 1934 auf dem Reichsparteitag produzierte der österreichische Postkartenmaler Adolf H.: „Eure vermeintliche gotische Verinnerlichung paßt schlecht in das Zeitalter von Stahl und Eisen, Glas und Beton, von Frauenthönheit und Männerkraft, von hochgehobnem Haupt und trotzigem Sinn . . . Unsere Sprache wird in hundert Jahren die europäische Sprache sein. Die Länder des Ostens, des Nordens wie des Westens werden, um sich mit uns verständigen zu können, unsere Sprache lernen. Die Voraussetzung dafür: An die Stelle der gotisch genannten Schrift tritt die Schrift, welche wir bisher die lateinische nannten . . .“⁷

Sieht auch jeder, was da oben falsch ist? Im letzten Absatz inhaltlich alles.

Aber vom Schriftbild fällt jedem, der sich mit Schriften beschäftigt hat auf, dass die „s“ nicht stimmen. Die „s“ am Ende von Wörtern (oder Silben) sind nicht richtig. In den deutschen Schriften unterscheidet man zwischen einem Lang-s `ſ` und einem Schluß-s `ß`, das durch `s:` eingegeben wird. Ein kleines geniales Beispiel habe ich bei Berndt⁸ auf Seite 312 gesehen: ein kleiner Kreis ist ein Kreischen (ein Kreis-chen), während zu Beginn der 60er-Jahre des 20. Jahrhunderts in Beatles-Konzerten immer ein Kreischen (ein Gekreische) zu hören war. Während bei dem kleinen Kreis das ‚s‘ zur ersten Silbe gehört und nach dem ‚s‘ eine neue Silbe anfängt, gehören bei dem Lärm `s,c` und `h` in die selbe Silbe. Die geometrische Figur wird also in Gotik `Kreischen`, der Lärm `Kreisſchen` geschrieben.^{9,10}

Hier kommt also nochmals der erste obige Absatz, aber richtig gesetzt:

Alte deutsche Schrift wird heute fast nur noch als Zielschrift gelehrt. Die Geschichte der deutschen Schriften in der jüngsten Vergangenheit ist schon eigenartig. Wie man vermuten könnte, war in der schlimmsten Zeit der deutschen

⁶Da ich normalerweise an einem Nicht-Windoofs-Rechner sitze, kann ich das nicht ausprobieren, wie das dort gemacht wird.

⁷Völkischer Beobachter Nr. 250 vom 7. September 1934, zitiert nach: <http://de.wikipedia.org/wiki/Normalschifterlass#Normalschifterlass>

⁸`berndt-ldtemd2008`

⁹gar nicht schlecht, wenn eine Schrift den Sinn des Geschriebenen verdeutlicht

¹⁰Auch hier hätten die Experten für die Rechtschreibereform einen Anstoß für ein bißchen mehr Nachdenken bekommen dürfen: ein Text wird einmal geschrieben, aber meistens viel öfter gelesen. Viele der durch dieses Gremium abgeschafften Regeln hatten zum Ziel, einen Text lesbarer zu machen; jetzt bleibt das Auge an sehr unschönen und unübersichtlichen Buchstabenkombinationen hängen (beispielhaft sei die Konsonanten-Vervielfachung genannt), so dass der Lesefluß unterbrochen wird.

Und jetzt sollten diese Experten die Aufsätze von Schülern korrigieren dürfen, damit sie sehen, ob die von ihnen angestoßenen Änderungen das Schreiben verbessert haben.

12. Was sonst noch so alles geht mit L^AT_EX

Geschichte, im Nationalsozialismus, alles Deutsche besonders gefördert, also auch Schriften wie Gotik oder Sütterlin. Und dann waren es diese besonders deutschen Deutschen, die per Gesetz diese Schriften abschafften.

Die Sütterlin ist hierbei noch etwas Besonderes: es ist eine Schreibschrift. Damit ist klar, dass Sütterlin nur zu Demonstrationszwecken in einem Druckwerk zu finden sein sollte. Also demonstriere ich:

Alle Sütterlin Schrift wird für die fast nur noch als Zierchrift gesehen. Die Schrift der Sütterlin Schrift in der jüngsten Vergangenheit ist sehr signifikant. Die man vor allem kennt, nur in der schlimmsten Zeit der Sütterlin Schrift, im Nationalsozialismus, alle Sütterlin besonders gefördert, also auch Schrift wie Gotik oder Sütterlin. Und dann waren es diese besonders Sütterlin Schrift, die per Gesetz diese Schrift abschafften.

12.4. Initialen

12.4.1. ...zuerst in alter deutscher Schrift

Texte, die in einer deutschen Schrift gesetzt sind, haben oft noch eine besonders schöne Verzierung, eine Initiale zu Beginn eines Kapitels. Dazu benötigt man wieder ein spezielles Paket, nämlich `lettrine`. Der `lettrine`-Befehl hat einige optionale Parameter, die die Größe der Initiale und Positionierung und Abstände steuern. Für Initialen gibt es verschiedene Muster, die unter Linux im Verzeichnis `/usr/share/texmf/doc/fonts/initials` gespeichert sind. Das unten stehende Beispiel benutzt den normalen Buchstaben der aktuellen Schrift.

Alte deutsche Schrift wird heute fast nur noch als Zierschrift gesehen. Die Geschichte der deutschen Schriften in der jüngsten Vergangenheit ist schon eigenartig. Wie man vermuten könnte, war in der schlimmsten Zeit der deutschen Geschichte, im Nationalsozialismus, alles Deutsche besonders gefördert, also auch Schriften wie Gotik oder Sütterlin. Und dann waren es diese besonders deutschen Deutschen, die per Gesetz diese Schriften abschafften.

Das Paket `yfonts` liefert auch noch einige Initialen, deren Handhabung besonders einfach ist:

Alte deutsche Schrift wird heute fast nur noch als Zierschrift gesehen. Die Geschichte der deutschen Schriften in der jüngsten Vergangenheit ist schon eigenartig. Wie man vermuten könnte, war in der schlimmsten Zeit der deutschen Geschichte, im Nationalsozialismus, alles Deutsche besonders gefördert, also auch Schriften wie Gotik oder Sütterlin. Und dann waren es diese besonders deutschen Deutschen, die per Gesetz diese Schriften abschafften.

12.4.2. Aber es geht auch bei beliebigen Schriften

Also funktioniert das auch mit der Butter- und Brot-Schrift:

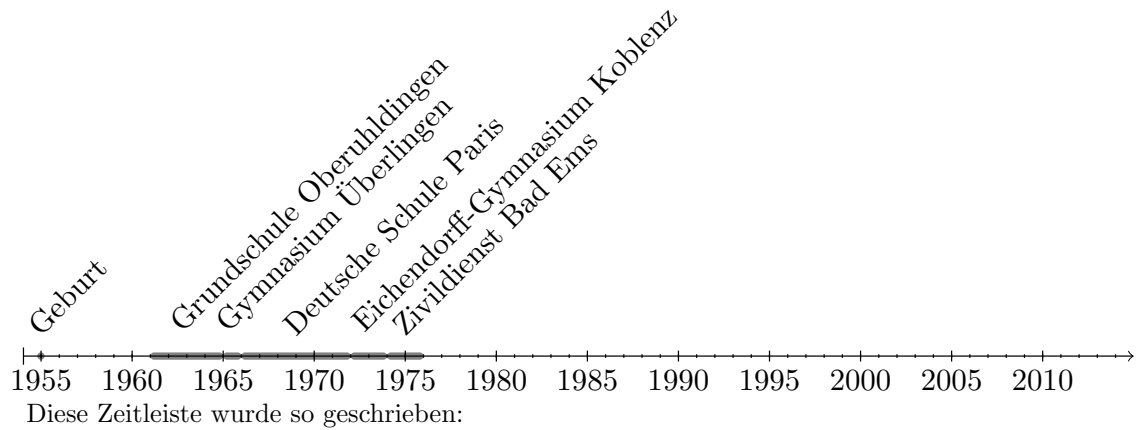
ALTE deutsche Schrift wird heute fast nur noch als Zierschrift gesehen. Die Geschichte der deutschen Schriften in der jüngsten Vergangenheit ist schon eigenartig. Wie man vermuten könnte, war in der schlimmsten Zeit der deutschen Geschichte, im Nationalsozialismus, alles Deutsche besonders gefördert, also auch Schriften wie Gotik oder Sütterlin. Und dann waren es diese besonders deutschen Deutschen, die per Gesetz diese Schriften abschafften.

Der Befehl für diesen Einzug benutzt die Kapitälchen aus dem Paket `soul`. Der gesamte Befehl sieht so aus:

```
\lettrine[lines=2, lhang=0.1, loversize=0.2, findent=1ex]{A}{\caps{lte}}
```

12.5. Zeitleisten

Zeitleisten werden mit dem Paket `chronology` erstellt und sehen so aus:



```
\begin{chronology}[5]{1954}{2014}{0.5cm}{0.95\textwidth}
  \event{\decimaldate{29}{12}{1954}}{Geburt}
  \event[1961]{1965}{Grundschule Oberuhldingen}
  \event[1965]{1966}{Gymnasium Überlingen}
  \event[1966]{1972}{Deutsche Schule Paris}
  \event[1972]{1974}{Eichendorff-Gymnasium Koblenz}
  \event[1974]{\decimaldate{31}{12}{1975}}{Zivildienst Bad Ems}
\end{chronology}
```

12.6. Lebenslauf

Für das Schreiben von Lebensläufen stehen verschiedene Pakete zur Verfügung:

- `CurVe`
- `currvita`
- `cv`
- `europecv`
- `moderncv`

Ein Lebenslauf in `moderncv` sieht etwa so aus:

Martin Schimmels

Bla-Straße
72xxx R...
+49 707x yy zz
martin.mschimmels@blub.de
www.meinchaos.de

Persönliche Daten

Geburtsdatum **xx.yy.1954 in Mainz.**
Familienstand **geht niemanden was an.**
Nationalität **deutsch.**

Schule und Ausbildung

1961–1965 **Grundschule**, Oberuhldingen.
1965–1966 **Gymnasium**, Überlingen.
1966–1972 **Gymnasium**, *Deutsche Schule*, Paris.
1972–1974 **Gymnasium**, *Eichendorff-Gymnasium*, Koblenz, Allgemeine Hochschulreife.

Abschlußarbeit

Titel *Die Relevanz des Irrelevanten unter besonderer Berücksichtigung der Mondphasen*
Betreuer Prof. Semmelweis

Berufliche Erfahrungen

Berufliche Tätigkeiten

1984–1985 **Müllwerker**, *Stadt*, Txxxxx.
1985–1986 **Anwendungsprogrammierer**, *Große Firma*, Tübingen und München.
Verschiedene Projekte bei diversen Firmen und Behörden

Nebenberufliche Tätigkeiten

1980–1986 **Trainer**, *ASV Pxxxx*.
Volleyball Damen- und Herrenmannschaft
1990–1992 **Trainer**, *TuS Obbbbbb*.
Volleyball Damen- und Herrenmannschaft

Sprachen

Deutsch **Muttersprache**
Französisch **sehr gut in Wort und Schrift**
Englisch **gut in Wort und Schrift**

*6-jähriger Auslandsaufenthalt
hat man halt gelernt*

Computer Kenntnisse

Programmierung
Großrechner

Programmierung
PC

Textbearbeitung
Buchsatzsystem \LaTeX

Hobbies

Volleyball Spieler und Trainer in verschiedenen Vereinen

Gitarre nur für den Hausgebrauch

Motorrad Urlaubsreisen ins nähere Ausland

Martin Schimmels
Bla-Straße
72xxx R...
+49 707x yy zz
martin.mschimmels@blub.de
www.meinchaos.de

13. Dezember 2014

Stadt Hzzzzzzz

Personalwesen

Marktplatz

71xxx Hzzzzzzz

Sehr geehrte Damen und Herren,

Yes I will.

Mit freundlichen Grüßen

Martin Schimmels

12. Was sonst noch so alles geht mit L^AT_EX



















Eine nette Anleitung von Uwe Ziegenhagen findet man im Netz unter <http://www.dante.de/events/Archiv/dante2011/programm/vortraege/lebenslauf.pdf>

12.7. Smileys

!!! TO DO !!!

Normale Smileys: ☺ ☻

Smileys der tikz-Lib:

		
basic	sad	neutral
		
small smile	big smile	confused
		
sexy	look up	look down
		
angry	look left	look right
		
blush	L ^A T _E X	Tikz
		
martian	devilish	almost crying

12.8. Sonderzeichen

Die Angabe von Temperaturen auf der Celsius-Skala wird im normalen Text mit `\textcelsius` geschrieben. Dabei ist zu beachten, dass hinter dem Symbol für Grad Celsius Leerraum gelassen werden sollte, wenn Text folgt. Um `\textcelsius` zu benutzen, muss das Paket `textcomp` eingebunden werden.

Im Mathematik-Modus wird eine solche Angabe durch `$ 50^\circ\text{C}` geschrieben.

A. GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright ©2000, 2001, 2002, 2007, 2008

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document “free” in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or non-commercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of “copyleft”, which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The “Document”, below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as “you”. You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A “Modified Version” of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A “Secondary Section” is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of

A. GNU Free Documentation License

the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, \LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".)

To “Preserve the Title” of such a section when you modify the Document means that it remains a section “Entitled XYZ” according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.

2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document’s license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy

A. GNU Free Documentation License

will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

1. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
2. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
3. State on the Title page the name of the publisher of the Modified Version, as the publisher.
4. Preserve all the copyright notices of the Document.
5. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
6. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
7. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
8. Include an unaltered copy of this License.
9. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.

10. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the “History” section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
11. For any section Entitled “Acknowledgements” or “Dedications”, Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
12. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
13. Delete any section Entitled “Endorsements”. Such a section may not be included in the Modified Version.
14. Do not retitle any existing section to be Entitled “Endorsements” or to conflict in title with any Invariant Section.
15. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version’s license notice. These titles must be distinct from any other section titles.

You may add a section Entitled “Endorsements”, provided it contains nothing but endorsements of your Modified Version by various parties — for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled “History” in the various original documents, forming one section Entitled “History”; likewise combine any sections Entitled “Acknowledgements”, and any sections Entitled “Dedications”. You must delete all sections Entitled “Endorsements”.

6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an “aggregate” if the copyright resulting from the compilation is not used to limit the legal rights of the compilation’s users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document’s Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled “Acknowledgements”, “Dedications”, or “History”, the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License “or any later version”

A. GNU Free Documentation License

applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

11. RELICENSING

“Massive Multiauthor Collaboration Site” (or “MMC Site”) means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A “Massive Multiauthor Collaboration” (or “MMC”) contained in the site means any set of copyrightable works thus published on the MMC site.

“CC-BY-SA” means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

“Incorporate” means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is “eligible for relicensing” if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright ©YEAR YOUR NAME Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled “GNU Free Documentation License”.
--

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the “with ... Texts.” line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

B. Verzeichnisse

Abbildungsverzeichnis

3.1. Der Editor SciTE	23
3.2. Die IDE TeXMaker	25
3.3. Die IDE Kile	27
5.1. detexify!!	40
7.1. Vanitas vanitatis	77
7.2. ...et omnia vanitas	79
9.1. So sieht die Titelseite aus	89
9.2. So sieht die Titelseite aus	91

Listings

4.1. Minimaler Vorspann	30
4.2. Vollständiger Rahmen	31
5.1. Eine numerierte Liste	43
5.2. Eine geschachtelte numerierte Liste	44
5.3. Eine Liste mit Spiegelstrichen	45
5.4. Eine verschachtelte Liste mit Spiegelstrichen	45
5.5. Beschreibungsliste	46
5.6. Eine geschachtelte numerierte Liste mit neuer Gestaltung	47
5.7. Tabelle mit tabbing	50
5.8. tabbing mit fehlerhafter Musterzeile	51
5.9. Tabelle mit 3 Spalten	52
5.10. Tabelle mit fester Spaltenbreite	52
5.11. Tabelle mit waagrechten Strichen	53
5.12. Tabelle mit senkrechten Strichen	54
5.13. Horizontale Linien	55
5.14. Stützen	56
5.15. Einträge über mehrere Spalten	56
6.1. Einfacher Mathematik-Modus	60
6.2. Equation-Umgebung	64
6.3. Mathematische Sätze	65
6.4. Neuer Theoremstil	66
6.5. Kommutatives Diagramm mit AMSmath	70
6.6. Einbinden des Pakets	71
6.7. Kommutatives Diagramm mit pb-diagram	71
7.1. Figure-Umgebung	75
7.2. Bildüberschrift	75
7.3. Bildunterschrift	75
7.4. Floatfigure-Umgebung	77
7.5. Wrapfigure-Umgebung	79
9.1. Satz einer Titelseite	88
9.2. Satz einer Titelseite mit KOMA	90
10.1. Schlaue Sprüche	99
10.2. Schlaue Sprüche: Variante	99

Listings

11.1. Glossar-Eintrag	103
11.2. Eintrag ins Abkürzungsverzeichnis	103
11.3. Einbinden der Glossar-Umgebung	104
11.4. Import von Glossar und Abkürzungsverzeichnis	104
11.5. Drucken des Abkürzungsverzeichnisses	104
11.6. Umgebung für bibliography	105
11.7. Handgestrickte Literaturangaben	106
11.8. Beispiel für Literatureinträge	108
11.9. Befehle für die Einbindung des Literaturverzeichnisses	109
11.10 Befehle im Vorspann für biblatex	109
12.1. Vorspann von beamer	112
12.2. Titelseite in beamer	113
12.3. Inhaltsverzeichnis in beamer	113
12.4. Eine Folie	114
12.5. Einfacheres Einblenden	114
12.6. Hervorhebung des aktuellen Punktes	115
12.7. Notensatz mit Lilypond	117

Listings

Listings

Glossar

CD Compact Disc 21, 144

DANTE ist das Akronym für Deutsche Anwendervereinigung T_EX . Der Zweck von DANTE „ist die Betreuung und Beratung von T_EX-Benutzern im gesamten deutschsprachigen Raum. Dazu gehört die Beratung sowohl mittels WWW-Server, E-Mail als auch „gelber“ Post, was die Anschaffung, Implementierung und Lösung von Anwendungsproblemen mit T_EX angeht, das Verteilen von Software an Mitglieder und die Informationsübermittlung dessen, was in der T_EX-Welt geschieht. Außerdem werden Entwicklungen im Bereich von T_EX, L^AT_EX, ConTeXt, LuaTeX, Metafont, BibTeX, ... national und international initiiert, gefördert und koordiniert.“ 18, 144

DVI Device Independant 29, 144

EPS Encapsulated Postscript 73, 144

GIF Graphics Interchange Format 144

HTML Abkürzung für HyperText Markup Language. HTML ist die Seitenbeschreibungssprache, mit Hilfe derer Internet-Seiten geschrieben (oder besser beschrieben) werden. HTML ist eine Auszeichnungssprache, das heißt, dass einzelne Elemente eines Textes mit Markierungen versehen werden, die dann vom Browser in einer bestimmten Weise dargestellt werden. Darin ähnelt HTML L^AT_EX ... aber wenn man die Mächtigkeit von L^AT_EX mit HTML vergleicht, bemerkt man, dass L^AT_EX um einiges umfangreicher ist. Aber das ist ja auch klar, denn das Anwendungsgebiet von L^AT_EX ist auch komplexer. 29, 144

JPG Joint Photographic Experts Group 73, 144

Knuth Donald E. Knuth, amerikanischer Mathematiker, geboren 1938, wurde weniger durch seine mathematischen Leistungen sondern mehr durch seine Arbeiten auf dem Gebiet der Informatik bekannt. Sein wahrscheinlich wichtigster Beitrag in diesem Bereich ist die Konzeption des Buchsatzsystems T_EX. Im Zusammenhang damit hat er auch ein Programm-System zur Entwicklung von Zeichensätzen, den MetaFont, beschrieben.

Ein besonders schönes Ergebnis seiner Arbeit ist das „3:16-Projekt“, in dem er ausgehend vom Johannesevangelium Kapitel 3 Vers 16 andere „Kapitel 3 Vers 16“-Textstellen der Bibel in verschiedenen Darstellungen in einem Buch zusammengefasst hat. 17, 144

Glossary

Lamport Leslie Lamport, amerikanischer Mathematiker und Informatiker, geboren 1941, ist in diesem Text zu erwähnen, weil er ein Makropaket zu $\text{T}_{\text{E}}\text{X}$ entwickelt hat, das $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, in dessen Namen er ganz bescheiden die ersten beiden Buchstaben seines Nachnamens untergebracht hat. 17, 144

Linux ist ein Unix-ähnliches, freies Betriebssystem. Es wurde von Linus Thorvalds begonnen, der nur die Multitasking-Fähigkeiten eines uralten Intel-Prozessors testen wollte. Dazu schrieb er einen minimalen Betriebssystem-Kern, den er ins Internet stellte, und, oh Wunder, hunderte, tausende Programmierer fanden das so spannend, dass sich daraus ein Programmpaket entwickelte, das seinesgleichen sucht. 21, 144

MS Microsoft 144

PDF Portable Document Format 29, 31, 144

PNG Portable Network Graphics 73, 144

PS Postscript 144

Umgebung Eine Umgebung ist ein zusammengehöriger Bereich eines $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Dokuments mit gemeinsamen Eigenschaften. 30, 31, 144

Index

`LATEX`-Befehl, 32
`LATEX`-Befehle
 `\LaTeX`, 41
 `\`, 36
 `\author`, 89
 `\begin{document}`, 33
 `\begin{math}`, 62
 `\bfseries`, 96
 `\caption`, 77
 `\chapter`, 99
 `\copyright`, 41
 `\cos`, 64
 `\date`, 89
 `\documentclass`, 32
 `\em`, 44, 95
 `\end{document}`, 33
 `\flqq`, 43
 `\footnote{}`, 85
 `\frac{}{}`, 62
 `\glqq`, 43
 `\includegraphics`, 76
 `\int`, 64
 `\listoffigures`, 76
 `\ln`, 64
 `\makeindex`, 103, 104
 `\maketitle`, 90
 `\mbox`, 64
 `\newglossaryentry`, 105
 `\newtheorem`, 67
 `\paragraph`, 99
 `\pm`, 63
 `\printglossary`, 106
 `\printindex`, 104
 `\protect`, 85
 `\publishers`, 92
 `\renewcommand{}{}`, 49
 `\section`, 99
 `\sin`, 64
 `\sqrt`, 63
 `\subject`, 92
 `\subsection`, 99
 `\subsubsection`, 99
 `\tan`, 64
 `\textregistered`, 41
 `\texttrademark`, 41
 `\thanks`, 89
 `\title`, 89
 `\titlehead`, 92
 `\underline`, 95
 `\usefonttheme`, 115
 `\usepackage`, 32
 `\usetheme`, 114
 `\verb`, 59
 `\addtolength`, 39
 `\flqq`, 43
 `\frqq`, 43
 `\glqq`, 43
 `\grqq`, 43
 `\listoffigures`, 103
 `\listoftables`, 103
 `\loadglsentries`, 106
 `\makeglossaries`, 106
 `\rule`, 39
 `\setlength`, 39
 `\underline`, 38
 `\~`, 38
 Akzente, 42
 Gliederungsbefehle
 `\chapter`, 99
 `\section`, 99

Index

- `\subparagraph`, 99
 - `\subsection`, 99
 - `\subsubsection`, 99
 - protect, 102
 - Schriftgrößen
 - `\Huge`, 96
 - `\Large`, 96
 - `\footnotesize`, 96
 - `\huge`, 96
 - `\large`, 96
 - `\normalsize`, 96
 - `\scriptsize`, 96
 - `\small`, 96
 - `\tiny`, 96
 - Sonderzeichen, 42
 - Zeilenwechsel, 36
- L^AT_EX-Distributionen, 23
- L^AT_EX-Dokumentklasse
 - beamer, 114
- L^AT_EX-Dokumentklassen
 - article, 99
 - book, 99
 - KOMA
 - scrartcl, 99
 - scrbook, 99
 - scrreprt, 99
 - letter, 99
 - report, 99
- L^AT_EX-Paket
 - amsmath, 69
 - amssymb, 69
 - amsthm, 67
 - float, 78
 - glossaries, 105
 - graphicx, 76
 - hyperref, 106
 - lettrine, 123
 - oldgerman, 121
 - wrapfig, 81
 - yfonts, 123
- L^AT_EX-Umgebung
 - description, 48
 - enumerate, 45
 - eqnarray, 65
 - figure, 75, 76
 - floatingfigure, 79
 - itemize, 47
 - multicols, 50
 - proof, 68
 - tabbing, 52
 - table, 75
 - tabular, 54
 - titlepage, 89
 - verbatim, 59
 - wrapfigure, 81
- Akzente, 42
- alte deutsche Schrift, 121
- Anführungszeichen, 43
- Ausgabeformat
 - dvi, 31
 - pdf, 31
- Auszeichnungssprache, 31
- Befehl, 32
- Beschreibungsliste, 48
- Bindestrich, 40
- Boxen, 37
- Deutsche Schrift, 121
- Distributionen, 23
- Dokumentklasse, 32
- dvi, 31
- Elefant, 103
 - indischer, 103
- Fehler, 34
- Fraktur, 43
- Gedankenstrich, 40
- Gleitobjekte
 - Positionierung, 77
- Hurenkinder, 102
- Initialen, 123
- KOMA, 32
- KOMA-Script, 17
- Längen, 38

Lebenslauf, 124
Leerzeichen, 38
Ligatur, 43
Liste, 45
 nicht-numerierte, 47
Listen
 mehrspaltig, 50

Matrizen, 71
Mengensymbole, 70

nicht-numerierte Liste, 47

Parameter, 32
pdf, 31
Proportionalschrift, 94

Quelltext, 31

Sütterlin, 43, 122
Satzspiegel, 36
Schrift
 deutsche, 121
 Proportional-, 94
Schusterjungen, 102
shell, 33
Silbentrennung, 35
Smileys, 128
Sonderzeichen, 40, 41
 Akzente, 42
 Grad Celsius, 128
Spiegelstrich, 45

Tabelle, 54
 einfach, 52
 Eintrag über mehrere Spalten, 58
 Spaltenmuster, 54
 Wiederholungen, 55

Umgebung, 32, 44
Unterschneidung, 43
usepackage, 32

Vergleichsoperatoren, 70
Vorspann, 32

Zahlenmengen, 71

Zeilenvorschub, 38
Zeitleisten, 124
zweispaltiger Satz, 86