
Schreiben wissenschaftlicher Arbeiten
mit L^AT_EX

— Eine kurze Einführung —

Tobias Baumbach, Matthias Dieke,
Michael Zipfel

29. März 2006

Inhaltsverzeichnis

1	Einleitung	3
1.1	Zielgruppe und Nutzen	3
1.2	T _E X und L ^A T _E X – was ist das	4
1.3	Was wird benötigt	4
1.4	Dokumentation	5
2	Erste Schritte	6
2.1	Dokumentaufbau	6
2.1.1	Dokumentenklasse	7
2.1.2	Pakete	7
2.1.3	Dokumentweite Einstellungen	7
2.2	Text schreiben	8
2.2.1	Satz- und Sonderzeichen	8
2.2.2	Text auszeichnen, Schrift verändern	9
2.3	Überschriften	11
2.4	Inhaltsverzeichnis und Titelseite	11
3	Marken, Zitate und Verweise	13
3.1	Fußnoten	13
3.2	Randnotizen	13
3.3	interne Verweise	14
3.4	Zitate	14
3.5	Quellenverweise	14
4	Spezieller Inhalt	16
4.1	Aufzählungen und Listen	16
4.2	Tabellen	17
4.3	Abbildungen	18
4.4	Gleitobjekte	19
4.5	Mathematische Formeln	20
5	Mehr Kontrolle	24
5.1	Trennung	24
5.2	Umbrüche	25
5.3	Seiten vergrößern	25
	Quellen und weiterführende Literatur	27

Kapitel 1

Einleitung

1.1 Zielgruppe und Nutzen

Diese kurze Einführung in den Gebrauch von \LaTeX ist für Leute gedacht, die eine wissenschaftliche Arbeit (wie z. B. eine Diplom- oder Masterarbeit) schreiben müssen. Gerade bei der Erstellung solcher Texte ist der Einsatz von \LaTeX angebracht und mit einigem Nutzen im Vergleich zu herkömmlichen Textverarbeitungsprogrammen verbunden.

\LaTeX ist (wie XML) eine so genannte Markupssprache. Das bedeutet, dass der Inhalt mit einfachen Befehlen – dem Markup – logisch strukturiert wird, das Layout des fertigen Dokuments aber an einer anderen Stelle festgelegt wird¹. Durch diese Trennung von Inhalt und Aussehen kann der selbe Inhalt für verschiedene Medien unterschiedlich aufbereitet werden. Die Strukturierung vereinfacht außerdem die automatische Erstellung von Verzeichnissen (Inhalts- und Literaturverzeichnis) und Verweisen.

Ein anderer Vorteil ist, dass sich der Autor besser auf die logische, inhaltliche Strukturierung der Arbeit konzentrieren kann. Da \LaTeX ein Klartextformat ist, das mit jedem beliebigen Editor gelesen und bearbeitet werden kann, besteht außerdem die Möglichkeit, die Daten von einem Versionskontrollsystem (RCS) verwalten zu lassen und so die Arbeit mehrerer Autoren an einem Dokument zu vereinfachen.

\LaTeX ermöglicht es, die Regeln der Typographie einzuhalten. Es wurde extra zum Setzen „schöner Texte“ entwickelt² und vereinfacht so nicht nur das Schreiben, sondern auch das Lesen. Ein wichtiger Unterschied im Vergleich zu Textverarbeitungsprogrammen ist die Erzeugung von sauberem Blocksatz mit optischem Randausgleich und einem ausgeglichenem Grauwert der Zeilen.³

Dieser Text soll einen schnellen und unkomplizierten Einstieg in \LaTeX ermöglichen. Daher werden nicht alle Befehle im Detail erklärt und die Arbeits-

¹Es existieren schon gute Layouts für die unterschiedlichsten Dokumenttypen, die von Profis entwickelt wurden.

²„ \TeX is a new typesetting system intended for the creation of beautiful books – and especially for books that contain a lot of mathematics.“ Donald E. Knuth, der Entwickler von \TeX

³DTP Programme wie Adobe InDesign können das auch, haben ihren Hauptnutzen aber bei Dokumenten mit viel gestalterischem Anteil. Normale Textverarbeitungsprogramme können keine komplizierten Textsatzalgorithmen verwenden, da eine Eingabe sofort eine Wirkung zeigen muss (WYSIWYG).

weise von \LaTeX erläutert⁴, sondern es werden nur die benötigten Befehle zum Erstellen einer Abschlussarbeit vorgestellt. Die vorliegenden 27 Seiten sollten ausreichen, um abschätzen zu können, ob \LaTeX das richtige Werkzeug für die anstehende Aufgabe ist, und um ohne Anfangsschwierigkeiten mit dem Schreiben beginnen zu können.

Im noch folgenden Teil von Kapitel 1 wird kurz die Geschichte von \TeX und \LaTeX beschrieben und erklärt, was man alles zum Einsatz dieses Textsatzsystems benötigt. In Kapitel 2 steht alles, was man zum schnellen Einstieg braucht. Die Satz- und Sonderzeichen, einfache Textformatierung und die ersten und wichtigsten Strukturierungsbefehle zum Erstellen von Überschriften werden erläutert und die Erstellung des Inhaltsverzeichnis erklärt. Kapitel 3 befasst sich mit Verweisen im weiteren Sinne. Thema sind neben Fuß- und Randnoten auch Textverweise innerhalb des Dokuments und die Erstellung und Nutzung eines einfachen Literaturverzeichnisses. In Kapitel 4 geht es um strukturierten Inhalt. Das umfasst Aufzählungen und Listen, Abbildungen und Tabellen, so genannte Gleitobjekte und den Satz von mathematischem Inhalt⁵. Das letzte Kapitel erklärt, wie man – nachdem der Inhalt geschrieben ist – dem Text durch Einfügen von kontrollierten Umbrüchen seine endgültige Form gibt und in einigen wenigen Fällen der automatischen Silbentrennung von \LaTeX auf die Sprünge hilft.

1.2 \TeX und \LaTeX – was ist das

\TeX ist eine Programmiersprache, die das genaue Setzen von Dokumenten ermöglicht. \TeX wurde bereits 1977 von Donald E. Knuth entwickelt, um die (Druck-)Qualität seiner eigenen mathematischen Bücher zu verbessern. Die Anwendung der recht komplexen Sprache erfordert jedoch eine lange Einarbeitungszeit. Um die Erstellung von Dokumenten zu vereinfachen entwickelte Leslie Lamport 1980 ein Makropaket, das viele der Lowlevel Befehle von \TeX zu Makros zusammenfasst und um einfach anzuwendende Funktionen erweitert. Dieses Makropaket ist \LaTeX . Im Laufe der Jahre sind viele Veränderungen und Ergänzungen an \LaTeX vorgenommen wurden. Etliche neue Pakete vereinfachen die Erstellung unterschiedlichster Dokumente, wie Bücher, Briefe oder Präsentationen. Die aktuelle Version ist seit 1996 $\text{\LaTeX}2_{\epsilon}$, es gibt auch eine Entwicklergruppe für \LaTeX 3.

1.3 Was wird benötigt

Sowohl \TeX als auch \LaTeX sind im Quellcode verfügbar und wurden in den letzten Jahren auf fast jedes Computersystem portiert. Auch komfortable Editoren, die die Eingabe der Befehle durch Syntaxhervorhebung und Befehlsvervollständigung vereinfachen, sind für die meisten Betriebssysteme kostenlos erhältlich. Solche Editoren sind z. B. LEd (LatexEditor, Windows) oder Kile (Linux, Macintosh). Die \LaTeX -Umgebung selbst kann man sich kostenlos von der Seite der CTAN (Comprehensive \TeX Archive Network) herunterladen. Für Windows

⁴Das ist schon in vielen anderen Büchern und Dokumenten erfolgt. Siehe Quellen und weiterführende Literatur.

⁵eine der besonderen Stärken von \TeX

beliebt ist MikTeX, in den meisten Linuxdistributionen ist schon eine L^AT_EX-Version enthalten. Je nach gewählter L^AT_EX-Distribution und Editor kann das druckfertige Dokument direkt aus dem Editor erzeugt und angesehen werden oder es ist ein Aufruf des L^AT_EX-Interpreters auf der Kommandozeile nötig.

1.4 Dokumentation

Viele der als Quellen aufgeführten Dokumente sind in einer lokalen L^AT_EX-Installation vorhanden. Sie liegen unter Linux unter `usr/share/doc/texmf/`. Die grundlegenden L^AT_EX-Dokumente liegen unterhalb dieses Verzeichnisses in `latex/general/` und `latex/base`. Wenn die angeführten Dokumente nicht in diesen (oder ähnlichen) Verzeichnissen vorhanden sind, können sie alle von der Seite der CTAN (www.ctan.org) heruntergeladen werden. Eine Suche nach bestimmten Dateien ist unter www.tug.org möglich. Die deutsche T_EX Usergroup ist der Dante e. V. Seine Webseite ist unter dante.ctan.org zu finden.

Kapitel 2

Erste Schritte

2.1 Dokumentaufbau

Ein \LaTeX -Dokument besteht aus einer so genannten Präambel, die die dokumentweiten Einstellungen enthält, und dem eigentlichen Inhalt. In der Präambel wird die Dokumentklasse festgelegt und eventuell weitere Zusatzpakete geladen. Außerdem können in diesem Teil kleinere Layoutänderungen vorgenommen werden, die das gesamte Dokument betreffen (siehe Listing 2.1).

Der Hauptteil enthält den strukturierten Inhalt. Er kann z. B. aus Kapiteln (\chapter), Abschnitten (\section , \subsection), Abbildungen, Tabellen, Listen und anderen Bestandteilen aufgebaut sein.

```
1 % Präambel
2 \documentclass[a4paper,10pt,twoside]{report}
3
4 \usepackage[latin1]{inputenc}
5 \usepackage[ngerman]{babel}
6 \usepackage[T1]{fontenc}
7
8 \author{Max Mustermann}
9 \title{Mein erstes Latex-Dokument}
10 \date{\today}
11
12 % Start des Inhalts
13 \begin{document}
14 \maketitle
15
16 \chapter{Einleitung}
17 \section{Anliegen}
18 Dieses Buch soll ...
19 \section{Hinweise}
20 ...
21 \end{document}
```

Listing 2.1: Grundstruktur eines \LaTeX -Dokuments

2.1.1 Dokumentenklasse

Der erste Befehl der Präambel – und damit von jedem L^AT_EX-Dokument – ist `\documentclass`. Mit diesem Befehl wird festgelegt, welcher Art ein Dokument ist und welche Befehle zum Auszeichnen der logischen Struktur benötigt werden. Im Standard-L^AT_EX sind u. a. die Klassen `article`, `book`, `letter`, `report` und `slides` definiert. Im deutschsprachigen Bereich sind als Erweiterung häufig noch die KOMAScriptKlassen nutzbar, die Layoutanpassungen an den deutschen Satz definieren. In dieser Einführung wird der Gebrauch der Klasse `report` erläutert, die das Erstellen von wissenschaftlichen Arbeiten vereinfacht. `article` ist, wie der Name schon sagt, für kleinere Artikel z. B. in Zeitschriften gedacht, während `book` für Bücher, `letter` für Briefe und `slides` für Folien am besten geeignet ist.

```
\documentclass [options]{style}
```

Welche Optionen erkannt werden, hängt von der verwendeten Klasse ab. Von den meisten (auch nicht Standard-) L^AT_EX-Klassen werden aber die Optionen `10pt`, `11pt`, `12pt` erkannt, die die zu verwendende Grundschriftgröße festlegt. Die Optionen `a4paper`, `a5paper`, `b5paper`, `letter` und `landscape` legen das Papierformat fest. Die Option `twoside` stellt den beidseitigen Druck ein. Optionen werden durch Kommata getrennt.

2.1.2 Pakete

Etliche Zusatzfunktionen, wie z. B. die automatische Silbentrennung für die deutsche Sprache, sind nicht nur für eine Dokumentklasse konzipiert. Solche Erweiterungen sind in Paketen zusammengefasst. Ein Dokument kann beliebig viele Pakete nutzen. Diese werden in der Präambel mittels

```
\usepackage [options]{package}
```

angegeben. Sind Optionen für mehrere Klassen gültig, können die Angaben kombiniert werden. `\usepackage[option1, option2]{package1, package2}` lädt die beiden angegebenen Pakete mit beiden Optionen. Zusätzlich werden allen Paketen die Optionen übergeben, die in der Dokumentklasse angegeben sind. Eine typische Kombination von Paketen ist in Listing 2.1 auf Seite 6 zu sehen. Das Paket `inputenc` legt den zu verwendenden Eingabezeichensatz fest (in diesem Falle `latin1` mit deutschen Umlauten) und das Paket `babel` mit der Option `ngerman` lädt die Silbentrennung der neuen deutschen Rechtschreibung (`german` würde die alten Trennregeln aktivieren).

2.1.3 Dokumentweite Einstellungen

Außer der Klasse und den Paketen lassen sich noch individuelle Einstellungen in der Präambel vornehmen, die dann für das gesamte Dokument gelten. Dazu gehören z. B. `\author`, `\title` und `\date`, die Informationen zu dem Dokument speichern, welche dann im späteren Verlauf beim Erstellen des Deckblattes genutzt werden. Andere Möglichkeiten sind Einstellungen, die die Kopf- und Fußzeilen betreffen oder auch der Zeilenabstand.

```
\linespread{1.25}\selectfont
```

Der Befehl `\baselinestretch` gibt an, mit welchem Wert der Zeilenvorschub (`\baselineskip`) multipliziert wird. Mit `\linespread` wird `\baselinestretch` und damit der Zeilenabstand verändert. Es ist allerdings zu beachten, dass \LaTeX den Zeilenvorschub schon ca. 20% größer festlegt als die Schriftgröße. Um bei einer Schriftgröße von 10pt (Zeilenvorschub ca. 12pt) einen anderthalb zeiligen Zeilenabstand zu erhalten, muss `\linespread` auf 1.25 gesetzt werden¹. Der Befehl `\selectfont` ist nötig, um den geänderten Zeilenabstand auch wirklich im Text zu verwenden. Sollte `\linespread` als Toplevelbefehl nicht funktionieren, kann auch `\baselinestretch` direkt verändert werden.

```
\renewcommand{\baselinestretch}{1.25}
```

Hier wird direkt der Befehl `\baselinestretch` mittels `\renewcommand` verändert. Das funktioniert immer, allerdings ist in \LaTeX generell ein höher abstrahierter Befehl einem direktem Weg vorzuziehen, da in höheren Befehlen immer Überprüfungen und Kompatibilitätsbedingungen enthalten sind, die man mit den direkten Befehlen übergehen würde.

2.2 Text schreiben

2.2.1 Satz- und Sonderzeichen

Da \LaTeX ursprünglich für die englische Sprache entwickelt wurde, sind auch oft gebrauchte Zeichen wie die deutschen Umlaute als Sonderzeichen zu berücksichtigen. Es besteht aber die Möglichkeit, den zur Eingabe und Ausgabe gewünschten Zeichensatz anzugeben. Dadurch ist das Einfügen von Sonderzeichen aus diesem Zeichensatz ohne weitere Beachtung möglich. Für deutsche Texte erledigen dies die beiden Pakete `inputenc` für die Eingabe und `fontenc` für die Ausgabe (siehe Listing 2.1 auf Seite 6). Eine andere Möglichkeit, auf besondere Zeichen zuzugreifen, besteht über \LaTeX -Befehle. Mit diesen ist es möglich, Akzente oder Wellen über Buchstaben zu platzieren und nicht auf der Tastatur vorhandene Zeichen ausgeben zu lassen (siehe Tabelle 2.1).

<code>\~o</code>	<code>\~w</code>	<code>ô</code>	<code>ô</code>	<code>\"o</code>	<code>\"w</code>	<code>ö</code>	<code>ö</code>	<code>\=o</code>	<code>\=w</code>	<code>ō</code>	<code>ō</code>
<code>\'o</code>	<code>\'w</code>	<code>ó</code>	<code>ó</code>	<code>\'o</code>	<code>\'w</code>	<code>ò</code>	<code>ò</code>	<code>\~o</code>	<code>\~w</code>	<code>õ</code>	<code>õ</code>
<code>\.o</code>	<code>\.w</code>	<code>ò</code>	<code>ò</code>	<code>\b o</code>	<code>\b w</code>	<code>o</code>	<code>o</code>	<code>\c o</code>	<code>\c w</code>	<code>q</code>	<code>q</code>
<code>\d o</code>	<code>\d w</code>	<code>o</code>	<code>o</code>	<code>\H o</code>	<code>\H w</code>	<code>ő</code>	<code>ő</code>	<code>\r o</code>	<code>\r w</code>	<code>ő</code>	<code>ő</code>
<code>\t ox</code>	<code>\t wx</code>	<code>ô</code>	<code>ô</code>	<code>\u o</code>	<code>\u w</code>	<code>ö</code>	<code>ö</code>	<code>\v o</code>	<code>\v w</code>	<code>ö</code>	<code>ö</code>
<code>\oe</code>	<code>œ</code>	<code>\OE</code>	<code>Œ</code>	<code>\ae</code>	<code>æ</code>						
<code>\AE</code>	<code>Æ</code>	<code>\aa</code>	<code>â</code>	<code>\AA</code>	<code>Å</code>						
<code>\o</code>	<code>ø</code>	<code>\O</code>	<code>Ø</code>	<code>\l</code>	<code>ł</code>						
<code>\L</code>	<code>Ł</code>	<code>\i</code>	<code>ı</code>	<code>\j</code>	<code>ĵ</code>						
<code>\ss</code>	<code>ß</code>	<code>\v i</code>	<code>ı</code>	<code>\"j</code>	<code>ĵ</code>						

Tabelle 2.1: Akzente und Sonderzeichen

¹(10pt + 20%) * 1.25 = 10pt * 1.5 = 15pt

Einige Zeichen haben bei der Eingabe in einem \LaTeX -Dokument eine besondere Bedeutung. Solche Zeichen kann man durch Voranstellen eines Backslash maskieren². Zu diesen Zeichen gehören $\$, \%, \&, \#, _ , \{$ und $\}$.

An- und Ausführungszeichen können bei Benutzung von `ngerman`³ durch `"` oder `\glqq{}` („) und `"` oder `\grqq{}` (“) erzeugt werden. Das einfache Ausführungszeichen kann durch die Eingabe von `'` (GraveAkzent) oder `\grq{}` erstellt werden, das einfache Anführungszeichen erhält man durch `\glq{}`. Ein Kurzzeichen für das einfache Anführungszeichen ist nicht vorhanden.

Striche unterschiedlicher Länge kann man durch das Aneinanderreihen von bis zu drei `-` (Minus) erzeugen. Dabei ist zu beachten, das der Geviertstrich (`---`) nur in angloamerikanischen Texten benutzt wird.

So wie es unterschiedliche Striche gibt, werden auch Leerstellen zwischen Zeichen und Worten unterschiedlich eingesetzt. Im deutsch- und französischsprachigen Raum wird das Leerzeichen zwischen zwei Sätzen nicht anders behandelt, als ein normaler Wortzwischenraum. Dieses Verhalten wird durch die Verwendung des `babel`-Paketes mit der Option `ngerman` automatisch eingestellt. Ansonsten kann die Sonderbehandlung von Satzzwischenräumen (breiter setzen) mit dem Befehl `\nonfrenchspacing` an-, und mit `\frenchspacing` ausgeschaltet werden. Ein `~` (Tilde) kann benutzt werden, um ein Leerzeichen einzufügen, an dem kein Zeilenumbruch erfolgen darf. `\,` wird verwendet, um ein schmales Leerzeichen zu setzen (bei Abkürzungen: `z.\,B.` wird zu z. B. im Gegensatz zu z. B.).

Auslassungspunkte können mit dem Befehl `\dots{}` gesetzt werden. Wann welche Satzzeichen und Abstände verwendet werden, hängt von den typographischen Konventionen eines Landes ab und kann sogar regional unterschiedlich sein. Die wichtigsten Regeln werden auch im Duden erläutert.

2.2.2 Text auszeichnen, Schrift verändern

Um einen Teil des Fließtextes besonders hervor zu heben, kann man ihn entweder fett drucken (aktiv), in Kapitälchen setzen (aktiv) oder kursiv schreiben (passiv)⁴. Es besteht auch die Möglichkeit, ein Wort zu sperren – d. h. die einzelnen Zeichen auseinander zu ziehen – das sollte aber nur von erfahrenen Setzern angewendet werden, da es den Grauwert und damit den Lesefluss erheblich stört. Auch der Einsatz von anderen Schriftarten und Schriftgrößen ist zu vermeiden. Je weniger unterschiedliche Schriften ein Text benutzt, um so besser ist er zu lesen.

```
\emph{text}
```

Durch `\emph` wird der in geschweiften Klammern angegebene Text hervorgehoben. Im normalen Text bedeutet das, dass der auszuzeichnende Teil kursiv gesetzt wird. Wenn innerhalb eines kursiven Textstücks `\emph` verwendet wird, schaltet \LaTeX wieder auf normale Schrift um.

²Durch das „maskieren“ werden Sonderzeichen vor dem Interpreter versteckt. Das maskierende Zeichen wird nicht ausgegeben, das maskierte Zeichen verliert jedoch seine besondere Bedeutung.

³Entweder als Paket mit `\usepackage{ngerman}` oder als Option für `babel` mit `\usepackage[ngerman]{babel}`

⁴aktive Auszeichnung bedeutet, dass der Leser schon beim Überfliegen der Seite den ausgezeichneten Teil bemerkt, passiv ausgezeichnete Worte fallen erst beim direkten Lesen auf

Befehl	Bereichsschalter	Name und Beispiel
<code>\textrm{text}</code>	<code>\rmfamily</code>	Antigua
<code>\textsf{text}</code>	<code>\sffamily</code>	Serifenlose
<code>\texttt{text}</code>	<code>\ttfamily</code>	Maschinenschrift
<code>\textmd{text}</code>	<code>\mdseries</code>	normal
<code>\textbf{text}</code>	<code>\bfseries</code>	fett, breiter laufend
<code>\textup{text}</code>	<code>\upshape</code>	aufrecht
<code>\textsl{text}</code>	<code>\slshape</code>	<i>geneigt</i>
<code>\textit{text}</code>	<code>\itshape</code>	<i>kursiv</i>
<code>\textsc{text}</code>	<code>\scshape</code>	KAPITÄLCHEN
<code>\textnormal{text}</code>	<code>\normalfont</code>	Grundschrift

Tabelle 2.2: Schriftstile

Außer `\emph` gibt es noch die Befehle des NFSS⁵, um die verwendete Schrift zu verändern. Um die Schriftfamilie, -serie oder -form auszuwählen, stehen die in Tabelle 2.2 auf Seite 10 aufgeführten Befehle zur Verfügung. Der Befehl verändert nur den im Argument angegebenen Text, der Bereichsschalter gilt bis zum Ende des ihn umschließenden Bereichs⁶ oder bis zum Befehl `\normalfont`.

Für die Schriftgröße gibt es weitere Befehle. Diese können als Bereichsschalter oder auch als normaler Befehl mit Argument verwendet werden und sind in Tabelle 2.3 aufgeführt. Die Befehle für Schriftstil und -größe können untereinander kombiniert werden.

Bereichsschalter	Beispiel
<code>\tiny</code>	winzige Schrift
<code>\scriptsize</code>	Indizegröße
<code>\footnotesize</code>	Fußnotengröße
<code>\small</code>	kleine Schrift
<code>\normalsize</code>	normale Schrift
<code>\large</code>	große Schrift
<code>\Large</code>	größere Schrift
<code>\LARGE</code>	sehr große Schrift
<code>\huge</code>	riesige Schrift
<code>\Huge</code>	am riesigsten

Tabelle 2.3: Schriftgrößen

⁵New Font Selection Scheme

⁶Ein Bereich kann jederzeit im Quellcode mit einem Paar geschweifter Klammern umschlossen werden. Diese werden nicht mit ausgegeben, sondern dienen zur Strukturierung des Textes. Auch Umgebungen begrenzen einen Bereich

2.3 Überschriften

Befehl	Name	Nummer
<code>\part</code>	Teil	-1
<code>\chapter</code>	Kapitel	0
<code>\section</code>	Abschnitt	1
<code>\subsection</code>	Unterabschnitt	2
<code>\subsubsection</code>		3
<code>\paragraph</code>	Paragraph	4
<code>\subparagraph</code>	Unterparagraph	5

Tabelle 2.4: Ebenenreihenfolge und interne Nummer

Wie schon erwähnt, muss man in \LaTeX die Struktur des Textes mit Markupbefehlen kennzeichnen. Die wichtigsten dafür sind Befehle, um die Überschriften der unterschiedlichen Ebenen auszuzeichnen. Von den Standardklassen bieten sowohl `report` als auch `book` die in Tabelle 2.4 aufgeführten Strukturierungsbefehle. Bei `article` fehlen die obersten zwei. Die Nummer wird benötigt, wenn man angeben will, bis zu welcher Ebene die Überschriften nummeriert bzw. in das Inhaltsverzeichnis eingetragen werden sollen. Die Befehle haben alle die gleiche Syntax.

```
\chapter{title}
\chapter*{title}
\chapter[short]{title}
```

Dabei ist `title` die Überschrift, wie sie im normalen Text erscheinen soll. Der Parameter `short` wird als Eintrag für das Inhaltsverzeichnis verwendet. Wird er nicht angegeben, so wird `title` auch dafür genutzt. Die Form mit Stern erzeugt nur eine Überschrift im Text. Diese wird weder nummeriert, noch erscheint sie im Inhaltsverzeichnis.

Um die Anzahl der Ebenen zu verändern, die nummeriert werden, muss man den Zähler `secnumdepth` neu setzen.

```
\setcounter{zähler}{wert}
```

Um bis zur Ebene der `\subsubsection` zu nummerieren, muss man also `\setcounter{secnumdepth}{3}` in der Präambel angeben.

2.4 Inhaltsverzeichnis und Titelseite

Die Strukturierung des Textes in Kapitel und Abschnitte erzeugt nicht nur automatisch Überschriften in der richtigen Größe und Schriftart, sondern \LaTeX erzeugt aus diesen Informationen automatisch ein Inhaltsverzeichnis. Dieses kann an beliebiger Stelle – meistens jedoch am Anfang eines Dokuments – eingefügt werden. Das geschieht durch den Befehl:

```
\tableofcontents
```

Im Normalfall werden Überschriften bis zur zweiten Ebene in das Verzeichnis übernommen. Wenn man die Tiefe ändern will, muss man den Zähler `tocdepth` neu setzen. Dabei wird die Nummer aus Tabelle 2.4 genutzt.

```
\setcounter{tocdepth}{3}
```

Durch setzen des Zählers auf den Wert „3“ werden also die Überschriften bis zur Ebene der `\subsubsection` in das Inhaltsverzeichnis übernommen.

Durch die Angabe von Metainformationen, wie z. B. Autor und Titel des Dokuments, kann L^AT_EX auch automatisch ein Deckblatt generieren. Dieses Deckblatt kann, genau wie das Inhaltsverzeichnis, an beliebiger Stelle im Dokument durch Angabe von `\maketitle` eingefügt werden.

```
15 \author{Max Mustermann}
16 \title{Mein erstes Latex-Dokument}
17 \date{\today}
18
19 \begin{document}
20 \maketitle
21 \tableofcontents
```

Listing 2.2: Metainformationen, Deckblatt und Inhaltsverzeichnis

Wenn einem die automatisch erzeugte Titelseite nicht gefällt, kann man in der `titlepage`-Umgebung ein eigenes Deckblatt gestalten. Innerhalb dieser Umgebung sind alle anderen Befehle verwendbar.

```
15 \begin{document}
16
17 \begin{titlepage}
18   \pagestyle{empty}
19   \Large{Der Titel der Arbeit}
20   \vspace{4em}
21   \small{Untertitel}
22 \end{titlepage}
23
24 \tableofcontents
```

Listing 2.3: die Umgebung `titlepage`

Der Befehl `\vspace` fügt einen vertikalen Abstand der angegebenen Länge ein (in Listing 2.3 vier mal die Breite eines „m“). Besonders nützlich ist in diesem Zusammenhang die elastische Länge `\stretch{nr}`. Alle als vertikale Länge genutzte `\stretch`-Aufrufe einer Seite teilen sich den verfügbaren Platz proportional zu ihrem Parameter auf. Wenn der freizulassende Platz am Beginn einer Seite steht, so muss `\vspace*{länge}` verwendet werden, da `\vspace{länge}` an einem Seitenumbruch ignoriert wird.

```
\vspace*{\stretch{3}}
Text nach 3/4 der Seite
\vspace*{\stretch{1}}
```

Kapitel 3

Marken, Zitate und Verweise

3.1 Fußnoten

Fuß- und Randnoten werden im Text markiert, der Text soll jedoch an anderer Stelle auftauchen. Für Fußnoten gibt es mehrere Befehle. Mit `\footnote` kann eine Fußnote im Text eingefügt werden. Dabei ändert der optionale Parameter `nr` die Standardnummerierung, `text` ist der Inhalt der Fußnote.

```
\footnote [nr]{text}
```

Der Befehl `\footnote` kann aber nicht in Überschriften oder Tabellen und Abbildungen verwendet werden. Um dennoch in solchen Textteilen eine Fußnote einfügen zu können, gibt es die Befehle `\footnotemark` und `\footnotetext`. Bei beiden kann die Nummer angegeben werden, mit der die Fußnote gekennzeichnet werden soll. Wird diese nicht angegeben, so nutzt `\footnotemark` die nächste Nummer nach der letzten Fußnote und `\footnotetext` bezieht sich auf die letzte Markierung. `\footnotetext` kann irgendwo im normalen Text hinter der Markierung stehen. Durch diese beiden Befehle ist es auch möglich, mehrere Markierungen im Text zu der selben Fußnote zu setzen.

```
\footnotemark [nr]  
\footnotetext [nr]{text}
```

3.2 Randnotizen

Randnotizen werden mit dem Befehl `\marginpar` gesetzt. Die Randnotiz erscheint mit der obersten Zeile bündig zu der Stelle, wo die Randnotiz angegeben wurde. Wenn nur der Pflichtparameter `right` angegeben wird, so wird die Notiz

- bei einseitigem Druck am rechten Rand
- bei zweiseitigen Druck am äußeren Rand
- bei mehrspaltigen Druck am nächsten Rand

gesetzt. Werden beide Parameter angegeben, so erscheint der Text von `left` links und der Text von `right` erscheint rechts.

```
\marginpar[left]{right}
```

3.3 interne Verweise

Verweise zu anderen Textstellen kann man mit dem Befehl `\label` im Text verankern, mittels `\ref` und `\pageref` kann man dann auf die Stellen verweisen. Dabei erzeugt `\ref` die Nummer des aktiven Elements, als `\label` aufgerufen wurde (siehe Abbildung 4.2 auf Seite 19, Zeile 15), `\pageref` erzeugt die Seitenzahl. Der verwendete Schlüssel muss im ganzen Dokument eindeutig sein.

```
\label{key}  
\ref{key}  
\pageref{key}
```

3.4 Zitate

Um Zitate besonders hervorzuheben, gibt es zwei Umgebungen. Die `quote`-Umgebung rückt den zitierten Text links und rechts ein, fügt aber keinen Absatzeinzug ein. Das geschieht bei der `quotation`-Umgebung, weshalb diese besser für lange Zitate geeignet ist.

```
\begin{quotation} \end{quotation}  
\begin{quote} \end{quote}
```

3.5 Quellenverweise

Wenn – z. B. nach Zitaten – eine Quellenangabe erfolgen soll, so gibt es verschiedene Verweisschemata, nach denen diese Verweise im Text formatiert werden können. Das Standardformat von `LATEX` ist das numerische. Das bedeutet, dass im Literaturverzeichnis alle Einträge durchnummeriert werden und im Text dann die Nummer des Eintrags steht. Sollten andere Schemata gewünscht sein, so müssen dem entsprechende Pakete geladen werden. Solche Pakete sind z. B. `cite`, `natbib` und `jurabib`.

```
\cite[text]{key}  
\cite[text]{key1, key2, ...}
```

Der Markupbefehl, mit dem Quellenangaben im Text erfolgen können lautet `\cite`. Dieser nimmt als Pflichtargument die Zitierschlüssel eines oder mehrerer Bücher aus dem Literaturverzeichnis entgegen und kann zusätzlich Text übergeben bekommen, der dann hinter dem Verweis erscheint (eine Seitenangabe o. ä.).

Die eigentliche Literaturliste kann direkt im Dokument erzeugt werden. Das ist jedoch nur bei kleineren Arbeiten zu empfehlen, da der Wartungsaufwand bei größeren Publikationen nicht zu vertreten wäre. So kann bei einer internen Literaturliste das Format der einzelnen Einträge nicht einfach geändert werden. Daher ist bei umfangreicheren Literaturangaben die Verwendung von `BIBTEX`

zu empfehlen. `BIBTEX` erstellt das Literaturverzeichnis automatisch anhand der verwendeten Verweismarken aus einer Datenbank.

Zum Erzeugen des Literaturverzeichnisses bzw. der Bibliographie wird die Umgebung `thebibliography` verwendet. In ihr werden die Werke aufgelistet, auf die im Text Bezug genommen werden kann. Die einzelnen Einträge werden mit `\bibitem` eingefügt. Dabei ist `label` der Eintrag, der dann im Text an der Stelle des Verweises erscheint und `key` ist der Zitierschlüssel, der mit `\cite` verwendet wird. Zusammengehörige Informationen können innerhalb eines `\bibitem`-Eintrags mit `\newblock` voneinander getrennt werden.

```
\begin{thebibliography}{widestlabel}
\bibitem[label1]{key1} bibliographic information
\bibitem[label2]{key2} author \newblock year
    ...
\end{thebibliography}
```

Kapitel 4

Spezieller Inhalt – Struktur im Fließtext

4.1 Aufzählungen und Listen

Standart- \LaTeX kennt drei verschiedene Umgebungen um Aufzählungen und Listen zu setzen. Die Umgebung `itemize` erzeugt eine Liste, in der die einzelnen Elemente durch ein Zeichen gekennzeichnet werden, `enumerate` erzeugt dagegen eine Liste, bei der die Elemente durchnummeriert werden. Die dritte Umgebung ist `description`, die für Beschreibungen verwendet werden kann und einen beliebigen Text vor das Listenelement stellt. Alle Umgebungen können bis zu einer Tiefe von vier Ebenen ineinander verschachtelt werden.

```
1 \begin{itemize}
2 \item Umgebungen können verschachtelt werden
3   \begin{itemize}
4     \item je nach Tiefe werden andere Zeichen als
5       Anstrich benutzt. Diese können geändert werden
6     \item[+] wie hier zum Beispiel
7   \end{itemize}
8   \begin{enumerate}
9     \item die Listen können verschieden sein
10     \begin{enumerate}
11       \item tiefere Ebene
12       \item eins.zwei
13     \end{enumerate}
14   \end{enumerate}
15   \begin{description}
16     \item[\LaTeX:] ist sehr nützlich
17     \item[die Anwendung:] von diesem Textsatzprogramm
18                           ist überhaupt nicht schwer
19   \end{description}
20 \end{itemize}
```

Listing 4.1: Schachtel von verschiedenen Listenumgebungen

Die Einrückungen im Listing 4.1 sind, wie überall in \LaTeX , nur zur besseren Übersicht. Aufeinanderfolgende Leerzeichen im Quelltext werden zu einem einzigen Leerzeichen zusammengefasst. Mehrere Zeilenumbrüche (mit *Enter*) beenden einen Absatz. Der Textfluss und die Zeilenumbrüche mit eventuellen Silbentrennungen werden von \LaTeX selbst festgelegt und sind abhängig von dem zur Verfügung stehenden Platz. Das in Abbildung 4.1 auf Seite 17 gezeigte Ergebnis wurde zum Beispiel mit einer Breite von 70% der normalen Textbreite gesetzt.

- Umgebungen können verschachtelt werden
 - je nach Tiefe werden andere Zeichen als Anstrich benutzt. Diese können geändert werden
 - + wie hier zum Beispiel
- 1. die Listen können verschieden sein
 - (a) tiefere Ebene
 - (b) eins.zwei

\LaTeX : ist sehr nützlich

die Anwendung: von diesem Textsatzprogramm ist überhaupt nicht schwer

Abbildung 4.1: Das Ergebnis von Listing 4.1 auf Seite 16

4.2 Tabellen

Gerade für wissenschaftliche Dokumente sind Tabellen unerlässlich. Sie bieten eine einfache Möglichkeit, viele Daten übersichtlich zu präsentieren. Daher ist der Mechanismus von \LaTeX für den Tabellensatz mächtig und umfangreich. Zwei Dinge sind in \LaTeX im Zusammenhang mit Tabellen zu beachten – zum Einen das Layout und die eigentliche Erstellung der Tabelle und zum Anderen die Platzierung der Tabelle im Text.

Die Erstellung einer Tabelle erfolgt in der `tabular`-Umgebung. Der optionale Parameter `pos` gibt an, wie die Tabelle in Bezug auf die Grundlinie gesetzt werden soll. Dieser Parameter hat nur eine Auswirkung, wenn die Tabelle keinen eigenen Absatz bildet, sondern im Fließtext steht. Dann bedeutet `t`, dass die Tabelle mit der oberen Kante an der Grundlinie ausgerichtet wird, `b` richtet an der unteren Kante aus. Wird kein Parameter angegeben, so richtet sich die Tabelle zentriert aus. Der Pflichtparameter `cols` beschreibt die Spaltentypen und eventuelle Trennzeichen der Tabelle. Die `tabular*`-Umgebung kann zusätzlich noch einen Breitenparameter verarbeiten, `array` wird in mathematischen Umgebungen verwendet, um Formeln zu setzen.

```

\begin{array}{cols}                # \end{array}
\begin{tabular}[pos]{cols}         # \end{tabular}
\begin{tabular*}{width}[pos]{cols} # \end{tabular*}

```

Formatkürzel	Typ
<code>l</code>	Linksbündige Spalte
<code>r</code>	Rechtsbündige Spalte
<code>c</code>	Zentrierte Spalte
<code>p{breite}</code>	oben ausgerichteter Absatz der angegebenen Breite
<code> </code>	vertikale Linie zwischen den Spalten
<code>@{dekl}</code>	Unterdrückt den Spaltenabstand, nutzt <code>dekl</code> als Trenner
<code>*{anz}{kürz}</code>	<code>anz</code> Exemplare von <code>kürz</code>

Tabelle 4.1: Spaltendeklarationen

Der Inhalt einer Tabelle besteht aus Zeilen und diese wiederum aus Spalten. Ein Spaltenwechsel erfolgt mit `&`, eine Zeile wird mit `\\` oder `\tabularnewline` beendet. Soll zwischen den einzelnen Zeilen eine Linie stehen, so kann man diese nach dem Zeilenwechsel mit dem Befehl `\hline` einfügen (siehe Listing 4.2 auf Seite 19 für ein Beispiel).

4.3 Abbildungen

Da $\text{T}_{\text{E}}\text{X}$ deutlich älter ist, als die heute gängigen Grafikformate, gibt es auch leider keine geräteunabhängige Möglichkeit, Bilder und Grafiken in das $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ -Dokument einzubinden. Da jedoch in jedem $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ -System ein Treiber zum Drucken bzw. Verarbeiten der `dvi`-Datei vorhanden ist, können in diese Datei spezielle Befehle eingearbeitet werden, die dann bei der weiteren Verarbeitung die gewünschten Bilder erzeugen. Dafür existieren einfach anzuwendende Pakete und Befehle. Je nach verwendetem `dvi`-Treiber können verschiedene Formate eingebunden werden und die Bilder direkt aus dem $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ -Dokument heraus transformiert (z. B. gedreht und skaliert) werden.

Da aus der `dvi`-Datei PostScript erzeugt wird, ist die Verwendung von `.ps`- bzw. `.eps`-Formaten immer möglich¹. Es ist daher häufig am einfachsten, die gewünschten Bilder vor der Benutzung in $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ mit einem Konvertierungsprogramm nach EPS² umzuwandeln.

Es besteht die Wahl zwischen dem älteren `graphics`-Paket und einem neueren Paket namens `graphicsx`. Dieses Letztere hat eine einfachere Syntax und wird deshalb hier weiter beschrieben. Bei der Einbindung des Pakets muss als Option der zu verwendende `dvi`-Treiber angegeben werden, der die weitere Verarbeitung der Bilder übernimmt.

```
\usepackage[dvips]{graphicsx}
```

Der eigentliche Befehl zum Einbinden eines Bildes ist `\includegraphics`. Im optionalen Parameter kann eine SchlüsselWertListe³ mit zusätzlichen Optionen angegeben werden, der Pflichtparameter enthält den Namen der Bilddatei.

¹Außer bei der Verwendung von $\text{P}_{\text{D}}\text{F}_{\text{T}}\text{E}_{\text{X}}$. Dabei wird direkt PDF erzeugt und deswegen kann kein PostScript eingebunden werden.

²Encapsulated PostScript

³z. B. `[width=10mm, height=20mm, angle=30]`

Der Befehl `\begin{center}` startet einen zentrierten Bereich, in welchem die eigentliche Tabelle steht. Diese besteht aus einer rechtsbündigen Spalte und einer Spalte, die mehrzeilige Einträge enthalten kann und halb so breit ist wie der normale Text im Dokument. Durch `\caption{}` wird die Beschriftung des gleitenden Objekts gesetzt. Dieser Beschriftung wird automatisch der entsprechende Name der Gleitobjektumgebung⁴ und die passende Nummerierung vorgeangestellt. Der Befehl `\label{}` setzt eine Textmarke, auf deren Position man im Text Bezug nehmen kann.

Wenn man statt `\begin{table}` `\begin{figure}` benutzt, so ändert sich die Beschriftung von „Tabelle“ auf „Abbildung“. Außerdem wird das Gleitobjekt nicht in das Tabellenverzeichnis, sondern in das Abbildungsverzeichnis eingetragen. Ansonsten ist es ohne Auswirkung, ob man im Inhalt der Gleitobjektumgebung eine Tabelle (mit `\begin{tabular}... \end{tabular}`), eine Abbildung (mit `\includegraphics`) oder etwas anderes einfügt.

4.5 Mathematische Formeln

Die ursprüngliche Aufgabe von $\text{T}_{\text{E}}\text{X}$ war es, mathematische Texte zu setzen. Diese Fähigkeit hat $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ „geerbt“. Es gibt zwei verschiedene Möglichkeiten, Formeln und anderen mathematischen Inhalt zu setzen. Die Formeln (oder auch nur Variablenamen) können mitten in einer Textzeile (`math`) stehen oder in einer eigenen, abgesetzten Zeile. Bei der zweiten Variante wird noch zwischen einer nummerierten (`equation`) und einer Form ohne Nummern (`displaymath`) unterschieden. Als Kurzform für die `math`-Umgebung kann `\(Formel \)` oder auch `$ Formel $` verwendet werden, die Kurzschreibweise für `displaymath` ist `\[Formel \]`.

<code>\begin{math}</code>	<code>\end{math}</code>
<code>\begin{displaymath}</code>	<code>\end{displaymath}</code>
<code>\begin{equation}</code>	<code>\end{equation}</code>

Innerhalb einer solchen mathematischen Umgebung gelten andere Befehle, als im Textmodus. Exponenten werden mit `^{\text{exp}}` hochgestellt, Indizes mit `_{\text{ind}}` erstellt. Mit den selben Befehlen kann man auch obere und untere Grenzen festlegen. Wenn die `math`-Umgebung verwendet wird und die Formel dadurch in einer normalen Textzeile steht, so werden die Grenzen genau wie Exponenten und Indizes neben das Zeichen geschrieben (Abbildung 4.2 auf Seite 21).

Es gibt verschiedene Symbole und Zeichen, die ihre Größe abhängig vom Inhalt ändern. Klammern und Wurzelzeichen sind Beispiele dafür. Runde und eckige Klammern kann man direkt eingeben, geschwungene Klammern erhält man durch `\{` und `\}`. Andere Begrenzer erhält man durch die entsprechenden Befehle (z. B. `\lfloor x \rfloor` ergibt $[x]$). Setzt man ein `\left` bzw. `\right` vor die öffnenden und schließenden Klammern, so wird automatisch die richtige Größe gewählt. Das n -te Wurzelzeichen erhält man durch `\sqrt[n]` (Abbildung 4.3 auf Seite 21).

Andere größenveränderliche Zeichen sind waagerechte Klammern und Striche. Diese werden mit `\overbrace` und `\underbrace` bzw. `\overline` und `\underline` erzeugt. Der Text über und unter der Klammer wird wie Exponenten und Indizes erzeugt (siehe Abbildung 4.4 auf Seite 21).

⁴bei Benutzung von `german` ist das „Tabelle“

```

\int_{i=0}^{\infty}(x_1^{\{i/2\}})
\[\sum_{i=0}^{\infty}(x_1^{\{i/2\}})\]

```

$$\int_{i=0}^{\infty} (x_1^{(i/2)})$$

$$\sum_{i=0}^{\infty} (x_1^{(i/2)})$$

Abbildung 4.2: Summen, Integrale, Grenzen, Exponenten und Indizes

```

\left(\sqrt[3]{\left(x+y\right)+7}\right)

```

$$\left(\sqrt[3]{(x+y)}\right) + 7$$

Abbildung 4.3: Wurzeln und Klammern

Brüche werden mit `\frac` gesetzt. Für Binomialkoeffizienten gibt es den Befehl `\choose`. Um übereinander gesetzte Terme zu erhalten, nimmt man `\atop`. In Abbildung 4.5 auf Seite 22 ist ein Beispiel zu sehen.

Für Vektoren und Matrizen wird am Besten die Umgebung `array` verwendet (Abbildung 4.6 auf Seite 22). Sie funktioniert genau, wie die normale Tabellenumgebung (`tabular`).

Die `array`-Umgebung kann auch verwendet werden, um Gleichungssysteme aneinander auszurichten. Eine spezielle `array`-Umgebung mit der Spaltendefinition `{rc1}` ist schon in der Umgebung `eqnarray` definiert. Bei `eqnarray` ist eine Umschließende Matheumgebung schon impliziert. Die Variante mit Stern gibt keine Gleichungsnummern mit aus. (Siehe Abbildung 4.7 auf Seite 23.)

Für mathematische Formeln und Theoreme reicht ein Alphabet häufig nicht aus. Daher ist es möglich, mit Schriftbefehlen das gewünschte Alphabet auszuwählen. Wenn in der Präambel ein anderes Schriftpaket eingebunden wurde, ist es möglich, das nicht alle Varianten auch ein anderes Ergebnis liefern. In der Standardkonfiguration sind jedoch die Alphabete aus Tabelle 4.2 auf Seite 23 verfügbar. Die kaligraphischen Buchstaben sind nur als Großbuchstaben vorhan-

```

\left(\underbrace{(x+y)}_{>0}\right)+\overline{AB}

```

$$\underbrace{((x+y))}_{>0} + \overline{AB}$$

Abbildung 4.4: waagerechte Klammern und Striche

```

\[\frac{x^2-y^2}{x+y}=x-y\]
\[{a \choose b+c}\]
\[{x+y \atop w+z}\]

```

$$\frac{x^2 - y^2}{x + y} = x - y$$

$$\binom{a}{b+c}$$

$$x + y$$

$$w + z$$

Abbildung 4.5: Brüche und Binomialkoeffizienten

```

\begin{displaymath}
\mathbf{X} =
\left(\begin{array}{cc}
3 & 4 \\
7 & 1 \\
2 & 1
\end{array}\right)
\end{displaymath}

```

$$\mathbf{X} = \begin{pmatrix} 3 & 4 \\ 7 & 1 \\ 2 & 1 \end{pmatrix}$$

Abbildung 4.6: Vektoren und Matrizen

den. Zusätzlich kann noch das griechische Alphabet verwendet werden. So wird zum Beispiel $\delta\Delta\Omega$ durch die Befehle `$\delta\Delta\Omega$` erzeugt.

Allgemein bekannte Funktionen wie `sin` oder `log` werden nicht kursiv gesetzt, sondern aufrecht. Um das zu erreichen, nutzt man am besten die vorhandenen Befehle, die meist recht einleuchtende Namen haben (`\arcsin` , `\cotan` , `\ln` , `\gcd`). Daher bleibt hier eine vollständige Aufzählung aus. Im Literaturverzeichnis sind viele Einführungen in \LaTeX aufgeführt, die solche Übersichten enthalten. Eine fast vollständige ist in [1, Kapitel 4.5, S. 30-33] zu finden.

```

\begin{displaymath}
\begin{array}{r @{} l}
a & b \\
b & (c+d)
\end{array}
\end{displaymath}
\begin{eqnarray*}
x+y & = & 3 \\
2x+3y & = & 10
\end{eqnarray*}

```

$$\begin{array}{l}
a < b \\
b < (c + d)
\end{array}$$

$$x + y = 3 \quad (4.1)$$

$$2x + 3y = 10 \quad (4.2)$$

Abbildung 4.7: Gleichungssysteme

ABCabc	ABCabc
\mathbf{ABCabc}	ABCabc
ABCabc	ABCabc
\mathtt{ABCabc}	ABCabc
\mathcal{ABC}	<i>ABC</i>

Tabelle 4.2: verfügbare Alphabete im Mathematikmodus

Kapitel 5

Mehr Kontrolle

In \LaTeX berechnen mächtige Algorithmen die besten Stellen für einen Zeilen- oder Seitenumbruch. Dabei steht ein ausgewogenes Schriftbild und ein perfekt gesetzter Blocksatz im Vordergrund. Natürlich übersteigt es die Fähigkeiten jedes Programms, bestimmen zu können, ob ein Umbruch an dieser (optisch perfekten) Stelle auch semantisch zu vertreten ist¹. In solchen Ausnahmefällen und bei der Trennung von unbekanntem Worten muss mit Spezialbefehlen in den normalen Ablauf eingegriffen werden.

5.1 Trennung

Durch das Laden einer Sprachoption in der Präambel wird eine Datei mit Trennmustern für die angegebene Sprache geladen. In dieser Datei sind Buchstabenkombinationen und mögliche Trennung enthalten. Dadurch ist es möglich mit relativ wenigen Angaben ein Großteil aller Trennungen in der jeweiligen Sprache angegeben werden. Es ist aber immer möglich, dass \LaTeX zu einem bestimmten Wort nicht alle möglichen Trennstellen findet.

Eine erste Anlaufstelle ist die Fehlermeldung `Overfull \hbox` beim \LaTeX -Durchlauf. Nach dieser Fehlermeldung steht der Satzteil, der die Probleme verursacht hat und es werden die in Betracht gezogenen Trennstellen angegeben. Ansonsten kann man sich auch mit dem Befehl `\showhyphens{}` die bekannten Trennstellen anzeigen lassen². So gibt `\showhyphens{Trennstelle}` `Trenn-stel-le` aus.

Zusätzliche Trennstellen können bei Verwendung des Paketes `babel` mit `"-` direkt im Text eingefügt werden. Immer möglich ist die Angabe von ausschließlichen Trennstellen mit `\-`. Das Wort wird dann nur noch an den angegebenen Stellen getrennt³. Wenn ein Wort oft im Text vorkommt, so kann man die gewünschten Trennungen auch in der Präambel (vor `\begin{document}`) angeben.

```
\hyphenation{Gleit-objekt-umgebung
             Gleit-objekt-umgebungen
             zu-sammen-ge-setzt}
```

¹sind es Staub-ecken oder Stau-becken

²Die Ausgabe ist möglicherweise nur bei einem direkten Aufruf des \LaTeX -Interpreters auf der Kommandozeile zu sehen.

³`Gleit\objekt\umgebung` hat jetzt zwei mögliche Trennstellen.

Soll eine Trennung verhindert werden, so kann man den Textteil in eine Box setzen. Dadurch wird der Text in der Box auf alle Fälle in einer Zeile gesetzt. Es kann aber passieren, dass so die Zeile überläuft.

```
normaler Text \mbox{hier kein Umbruch} weiter
```

Worte, die einen Bindestrich enthalten, werden auch nur an diesem umgebrochen. Will man zusätzliche Trennungen erlauben, so muss man (wieder nur bei Verwendung von `babel`) `"=` an Stelle des Bindestrichs einfügen. Auch `"` lässt andere Trennstellen zu, wird aber an der so markierten Stelle umgebrochen, so erscheint kein Bindestrich am Zeilenende. Im Gegensatz dazu kann man mittels `"~` einen geschützten Bindestrich einfügen, an dem nicht umgebrochen werden darf.

5.2 Umbrüche

`LATEX` versucht immer einen ausgeglichenen Grauwert der Zeilen und einen sauberen Rand beim Blocksatz zu erhalten. Durch den komplexen Algorithmus, der dieses Verhalten steuert, kann es manchmal zu ungewöhnlichem Verhalten kommen. So ist es zum Beispiel möglich, das beim Löschen eines Wortes der Zeilenumbruch eher vorgenommen wird und der Absatz eine Zeile länger wird. Um solches Verhalten zu vermeiden, ist es manchmal am einfachsten, einen manuellen Zeilen- oder Seitenumbruch einzufügen oder explizit zu verhindern.

```
\linebreak
\nolinebreak
\pagebreak
\nopagebreak
```

Eine weitere Möglichkeit des Seitenumbruchs sind die Befehle `\clearpage` und `\cleardoublepage`. Der letzte Befehl sorgt außerdem dafür, das bei zweiseitigem Druck die nächste Seite eine rechte (ungerade) Seite ist, auch wenn dafür eine leere Seite ausgegeben werden muss.

Zeilenumbrüche verhindern kann man (wie schon in Kapitel 5.1 gezeigt) durch das Umgeben von Text mit einer `\mbox{}`. Soll aber nur an einem Wortzwischenraum nicht umgebrochen werden, so kann man ein geschütztes Leerzeichen einfügen. Diese gibt es in zwei Größen: `~` erzeugt ein geschütztes Leerzeichen normaler Breite, `\,` einen kleinen Zwischenraum, wie er z. B. zwischen Titel und Name oder zwischen den Zeichen von Abkürzungen verwendet wird⁴.

5.3 Seiten vergrößern

Wenn auch die manuellen Umbruchkorrektur nicht den erwünschten Erfolg hat, bleibt als letztes Mittel, einige Seiten (im Normalfall gegenüberliegende Seiten vor der problematischen Stelle) zu vergrößern oder zu verkleinern. Dabei sollte als Maß immer eine ganze Zeilenhöhe benutzt werden⁵. Dazu kann der Befehl

⁴z. B. wird durch `z.\,B.` erzeugt (zum Vergleich `z.B.`, `z.\,B.`, `z. B.` – `z.B.`, `z. B.`, `z. B.`)

⁵`\enlargethispage{-1\baselineskip}` verkleinert die Seite um eine Zeile

`enlargethispage` verwendet werden. Dieser Befehl muss zwischen zwei Absätzen auf den zu verändernden Seiten eingefügt werden. Dadurch wird dem Text der angegebene Platz mehr zur Verfügung gestellt.

```
\enlargethispage{size}  
\enlargethispage*{size}
```

Während die erste Variante ohne `*` die restliche Formatierung des Textes beibehält und einfach die Höhe des Textbereichs auf der Seite verändert, staucht oder streckt der zweite Befehl den verfügbaren Weißraum der Seite so weit es möglich ist.

Quellen und weiterführende Literatur

lokale elektronische Dokumente, oder von CTAN oder TUG

- [1] W. Schmidt et al.; *LaTeX 2_ε-Kurzbeschreibung*. Version 2.3, 2003
- [2] Leslie Lamport et al.; *LaTeX 2_ε – The macro package for TeX*. Edition 1.6, 1994
- [3] T. Oetiker et al.; *The Not So Short Introduction to LaTeX 2_ε*. Version 4.00, 2003
- [4] LaTeX 3 Project Team; *LaTeX 2_ε for authors*. 2001
- [5] Heiner Lamprecht; *LaTeX 2_ε – eine Einführung*. 2001
- [6] Manuela Jürgens; *LaTeX – eine Einführung und ein bisschen mehr...*. FernUniversität Hagen, 2000
- [7] Manuela Jürgens; *LaTeX – Fortgeschrittene Anwendungen*. FernUniversität Hagen, 1995
- [8] Christoph Bier; *Einige wichtige Typographische Regeln*. Version 1.4, 2005

Bücher

- [9] Leslie Lamport; *LaTeX: A Document Preparation System*. Addison-Wesley Publishing Company, 2. Ausgabe 1994
- [10] Frank Mittelbach, Michel Gossens; *Der LaTeX-Begleiter*. Pearson Studium, 2. Auflage 2005