

Praktisches \LaTeX

eine Einführung

Version 1.6

Helge Kreutzmann

Carsten Luckmann

Christian Jentsch

10. Juli 2001

Dieses Werk ist MICHÈLE K. gewidmet, zu deren Geburtstag es 1997 erstellt wurde. Über Kommentare und Anmerkungen unter `kreutzm@itp.uni-hannover.de` freue ich mich sehr. Dieses Buch darf frei kopiert, aber *nicht* ohne meine Zustimmung modifiziert werden. Die jeweils aktuelle Fassung ist über meine Homepage <http://www.itp.uni-hannover.de/~kreutzm> verfügbar.

Für Ergänzungen der Themen: Graphik unter DOS, WindowsTM 3.x, Windows 95TM, Windows 98TM, Windows NTTM sowie Ergänzungen/Tips zu Editoren bin ich besonders dankbar. Auch allgemeine Tips helfen!

Anmerkungen zum T_EXen unter Windows entweder an mich oder direkt an CHRISTIAN JENTSCH `jentsch@physik.uni-hannover.de`.

Diese Ausgabe ist gegenüber den vorhergehenden öffentlichen Ausgaben mehrfach ergänzt worden.

Ab Version 1.5 ist CARSTEN LUCKMANN (`luckmann@i.am`) Koautor, er ergänzte/erstellte mehrere Abschnitte, u. A. die Dokumentation zu den Paketen `longtable` und `floatflt` sowie die Kapitel zu $\mathcal{A}\mathcal{M}\mathcal{S}$ -L_AT_EX (6), BIB_TE_X (7), Indexerstellung (8) und pdf_TE_X (9).

Ach ja:

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Schriftstück berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, daß solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Inhaltsverzeichnis

1	Einleitung	9
1.1	Was will diese Dokumentation sein	9
1.2	Welche L ^A T _E X-Distributionen gibt es?	9
1.3	Was ist T _E X und was ist L ^A T _E X?	9
1.4	Kompatibilität und andere Probleme	10
2	Einen Text erstellen	13
2.1	Wie sag' ich's T _E X?	13
2.2	Die ersten Schritte	13
2.3	Umgebungen	14
2.4	Ein langes Dokument	15
2.5	Was will T _E X mir sagen?	17
2.6	Logisches vs. physisches Layout	17
3	Graphik	19
3.1	L ^A T _E X und Graphik – ein Überblick	19
3.2	Pakete rund um Graphik	21
3.2.1	Farbe	21
4	Tips und Tricks	23
4.1	Ich finde mein Zeichen nicht!	23
4.2	Floats – und was getan werden kann	23
4.2.1	Gleitobjekte im Fließtext	24
4.3	Tabellen	24
4.3.1	Lange Tabellen mit <code>longtable</code>	25
4.3.2	Lange Tabellen mit <code>supertabular</code>	25
4.3.3	Professionelle Tabellen mit <code>booktabs</code>	26
4.4	Numerierung und Bezüge	26
4.5	Folien	27
4.5.1	Echte Folien	27
4.5.2	Präsentationen mit dem Computer	28
5	Editoren und andere Hilfsmittel	31
5.1	L ^A T _E X und Unix (Linux)	31
5.2	L ^A T _E X und DOS	33
5.3	L ^A T _E X und Windows	33
5.4	Emacs und der T _E X-Modus	35
5.5	Mein PostScript-Bild ist nicht ok	37

5.6	Ein paar Worte zur Graphikkonvertierung	37
5.7	Hinweise zum Suchen von Paketen, Optionen etc.	37
6	$\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$	39
6.1	Gleichungsumgebungen	40
6.1.1	Einzelne Gleichungen	40
6.1.2	Gleichungsgruppen	41
6.1.3	Gleichungsteile	41
6.1.4	Weiteres	41
6.2	Weitere $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ -Befehle	42
6.2.1	Text	42
6.2.2	Matrizen	42
6.2.3	Klammern	42
6.2.4	Akzente	43
6.2.5	Weiteres	43
7	Literaturverzeichnisse	45
7.1	Literaturbezüge im Text	45
7.2	Lokale Literaturverzeichnisse	45
7.3	Literaturdatenbanken mit $\text{BIB}\mathcal{T}\mathcal{E}\mathcal{X}$	46
7.3.1	Das Datenbankformat	46
7.3.2	$\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ -Befehle für Literaturdatenbanken	49
7.3.3	$\text{BIB}\mathcal{T}\mathcal{E}\mathcal{X}$ -Durchlauf	50
7.3.4	Mehrere Literaturverzeichnisse	50
7.3.5	bib-Dateien veröffentlichen	51
8	Indexerstellung	53
8.1	Indexeinträge	53
8.2	Indexerzeugung und -ausgabe	54
9	$\text{pdf}\mathcal{T}\mathcal{E}\mathcal{X}$	57
9.1	Aufruf von $\text{pdf}\mathcal{T}\mathcal{E}\mathcal{X}$	57
9.2	Grundlegende $\text{pdf}\mathcal{T}\mathcal{E}\mathcal{X}$ -Befehle	57
9.3	Dokumenten-Information	58
9.4	Hyperlinks	58
9.4.1	Das <code>hyperref</code> -Paket	58
9.4.2	Manuelle Links	59
9.4.3	Inhaltsverzeichnis (manuell)	60
9.5	Einfach umschalten zwischen PDF- und DVI-Ausgabe	60
	Literaturverzeichnis	63

Abbildungsverzeichnis

2.1	Kurzes \LaTeX -Dokument	14
2.2	Ein langes Dokument	16
3.1	Eingebundenes png-Bild.	20
4.1	Eine gleitende Abbildung im aktuellen Absatz.	24
4.2	Eine normale Tabelle	25
4.3	Supertabular-Zusatzbefehle	26
5.1	bash-Skript zur \LaTeX -Automatisierung	32
5.2	Beispielhafte DOS-Batch-Datei für \LaTeX	34
5.3	Ergänzungen für <code>.emacs</code>	36

Tabellenverzeichnis

2.1	Die wichtigsten Befehlszeichen	14
2.2	Umgebungen	15
3.1	Ein Auszug aus den Optionen des includegraphics-Befehls	20
4.1	Mögliche Float-Parameter	24
4.2	Ausgabe des Tabellenbeispiels	25
4.3	Foil \TeX -Steuer-Befehle	28
5.1	Wichtige emacs-Tastenkombinationen	36
6.1	Optionen des Pakets <code>amsmath</code>	39
7.1	Literaturtypen von <code>BIB\TeX</code>	47
7.2	<code>BIB\TeX</code> -Felder	48
7.3	Die wichtigsten <code>BIB\TeX</code> -Stile	49
8.1	<code>makeindex</code> -Optionen	54
8.2	Die wichtigsten Schlüsselwörter für Formatänderungsdateien am Beispiel der Standardattribute	55
9.1	<code>hyperref</code> -Optionen	59
9.2	Argumente für den Parameter <i>appearance</i> des Befehls <code>\pdfdest</code>	59
9.3	Aktionen für Hyperlinks	59

Kapitel 1

Einleitung

1.1 Was will diese Dokumentation sein

Diese Dokumentation ist ein Howto, eine Einführung und ein Nachschlagewerk. Es behandelt sowohl die allerersten Schritte als auch Tips und Tricks. Darüber hinaus gibt es noch einige Hinweise im Umfeld von \LaTeX . Eigentlich besteht dieses Werk aus zwei Teilen. Der erste Teil wurde von mir erstellt. Er behandelt Grundlagen, Tips und Tricks und Graphik. Der zweite Teil ist [1] und kann frei von vielen ftp-Servern gezogen werden; er ist bei vielen \LaTeX -Distributionen schon dabei. Der zweite Teil enthält viele Details und Hinweise, die ich absichtlich nicht in den ersten Teil hineingenommen habe, insbesondere für grundlegende mathematische Eingaben sollte auf den zweiten Teil zurückgegriffen werden¹. Ich möchte darauf hinweisen, daß der zweite Teil nicht von mir stammt und ich keinen Kontakt mit den Autoren hatte. Der zweite Teil ist vollkommen unabhängig vom ersten lesbar, allerdings ist er auch knapper.

1.2 Welche \LaTeX -Distributionen gibt es?

Für viele Betriebssysteme gibt es mittlerweile \TeX -Distributionen. Der Unterschied liegt im Umfang der beigefügten Pakete. Diese erlauben es, bestimmte Probleme (z. B. Graphik) sehr einfach und elegant zu lösen. Weitere Pakete können aber in der Regel integriert werden, z. B. für neue Schriftsätze, spezielle Seitenlayouts, Satz von Noten usw. Im Internet sind die Pakete unter [ftp.dante.de](ftp:dante.de), [ftp.shsu.edu](ftp:shsu.edu) und vielen weiteren Rechnern erhältlich.

1.3 Was ist \TeX und was ist \LaTeX ?

Text einzugeben und zu drucken ist einfach, aber nicht umsonst ist der Beruf des Schriftsetzers mit einer mehrjährigen Ausbildung verbunden. Ein Dokument soll nicht (nur) schön aussehen, sondern es muß auch gut lesbar sein. Dabei gibt es viele Konventionen, die uns das Lesen stark vereinfachen.

¹Bei umfangreicherem mathematischen Inhalten empfiehlt sich $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX , das in Kapitel 6 beschrieben ist.

DONALD E. KNUTH entwickelte aus diesem Grunde \TeX . Damit lassen sich druckreif Texte und Formeln setzen. Auf \TeX aufbauend entwickelte LESLIE LAMPORT ein Makropaket, d. h. eine Routinensammlung, die viele Regeln bereits automatisch berücksichtigt. Sowohl \TeX als auch \LaTeX werden von ihren Originalautoren nicht mehr gewartet (s. dazu Abschnitt 1.4). Im weiteren werde ich nur auf \LaTeX eingehen.

Das Setzen inkl. Schriftauswahl, Schriftgrößenauswahl, Randbeachtung, Ligaturen, Positionierung von Tabellen und Graphiken uvm. wird von \LaTeX automatisch durchgeführt, allerdings läßt sich alles auf Wunsch auch selbst beeinflussen, was aber im Regelfall nicht notwendig ist.

\LaTeX wird über Befehle gesteuert, von der Form: „Summe mit Grenzen 0 und Unendlich“ oder „Tabelle mit fünf Spalten, drei linksbündig, zwei zentriert“. Am Anfang erscheint dies viel Tipparbeit, aber nach kurzer Zeit ist die hohe Effizienz bei der Eingabe spürbar. Insbesondere Formeln werden schnell und schön gesetzt.

1.4 Kompatibilität und andere Probleme

\LaTeX -Dokumente sind (fast) beliebig portierbar, d. h. \LaTeX läuft auf vielen Plattformen (verschiedene Unixe inkl. Linux, DOS, Macintosh, Amiga, Windows, OS/2 etc.) Ein \LaTeX -Dokument kann leicht von einem System auf ein anderes transferiert werden. So ist es z. B. (fast) problemlos möglich, auf einem PC unter DOS zu \TeX en und dann in der Uni von einer Unix-Workstation aus das Dokument zu bearbeiten und zu drucken². \LaTeX ist nicht perfekt. Viele Kleinigkeiten werden durch Zusatzpakete geregelt, aber auch einige große Punkte müssen noch verbessert werden.

Zum ersten ist das die Schriftverwaltung³. Ich werde hier *nicht* darauf eingehen, ich möchte nur anmerken, daß es in der derzeit aktuellen Version $\text{\LaTeX} 2_{\epsilon}$ ein neues Schriftverwaltungssystem gibt, das viele Mängel beseitigt hat, und damit dieses Problem deutlich entschärft wurde.

Das zweite und viel gravierendere Problem war Graphik. Aus diesem Grund steht im ersten Absatz nur „fast“. Leider war die Geräteunabhängigkeit bei Graphik nur sehr eingeschränkt gegeben. Mittlerweile läßt sich dies mittels spezieller Pakete nahezu vermeiden. Deshalb sollte auf jeden Fall Kapitel 3 sorgfältig studiert werden, bevor Graphik verwendet wird.

Ein Wort noch zu $\text{\LaTeX} 2_{\epsilon}$, $\text{\LaTeX} 2.09$ und $\text{\LaTeX} 3$. $\text{\LaTeX} 2_{\epsilon}$ ist die aktuelle und hier beschriebene Version. Sie sollte auf jeden Fall verwendet werden. Die ältere Version $\text{\LaTeX} 2.09$ funktioniert ebenfalls, allerdings sind hier die Schriftmängel und andere Probleme noch nicht behoben. $\text{\LaTeX} 3$ wird zur Zeit entwickelt. Eine interessante Übersicht über die Entwicklung ist auf <http://www.latex-project.org> dargestellt. Ich werde hier nicht darauf eingehen.

Inzwischen gibt es mehrere neue \LaTeX -Varianten. $\text{pdf}\text{\LaTeX}$ ermöglicht das Erstellen von PDF-Dokumenten einschließlich Kompression und Hyperlinks. Λ (bzw. Ω) erleichtert das Setzen vielsprachlicher Dokumente durch `unicode` und auch $\epsilon\text{-TeX}$ erweitert \TeX , hauptsächlich um schnell die Schreibrichtung

²Ich habe z. B. am Anfang völlig problemlos unter DOS ge \TeX ed und unter Linux gedruckt

³Oft als Fontverwaltung bezeichnet

links \leftrightarrow rechts ändern zu können. Von all diesen neuen \TeX s soll in diesem Dokument nur pdf \TeX in Kap. 9 näher vorgestellt werden.

Kapitel 2

Einen Text erstellen

2.1 Wie sag' ich's T_EX?

Am Anfang steht der Editor. Er sollte möglichst gut handhabbar sein und über Cut-and-Paste verfügen, Details dazu in Kapitel 5. In diesem Editor wird der Text geschrieben und abgespeichert, sagen wir unter `Beispiel.tex`. Jetzt wird der Text mit L^AT_EX übersetzt:

```
latex Beispiel.tex
```

Sind keine Fehler aufgetreten, so liegt jetzt die Datei `Beispiel.dvi` vor. `dvi` steht für „Device Independent“ und bedeutet die oben angesprochene Kompatibilität. Diese `dvi`-Datei kann nun einer Druckerei gegeben und dort mit 1600 dpi ausgedruckt oder auf dem alten 9-Nadel-Drucker zu Papier gebracht werden. Vor dem Druck kann natürlich das Ergebnis auf dem Bildschirm kontrolliert werden.

2.2 Die ersten Schritte

L^AT_EX kennt drei Arten von Zeichen¹: Normale Zeichen, Befehlszeichen und Sonderzeichen.

Normale Zeichen sind alle „normalen“ Buchstaben (a bis z und A bis Z) sowie die Ziffern und Satzzeichen. Normale Zeichen werden einfach über die Tastatur eingegeben.

Befehlszeichen steuern L^AT_EX. Das Befehlszeichen `\` sagt L^AT_EX, daß jetzt ein Befehl kommt, so heißt z. B. `\textbf`, daß fett gedruckt werden soll. Auch die Sonderzeichen entstehen so. So bedeutet `\alpha` schalte in den Mathemodus, drucke α und schalte wieder zurück. Für die deutschen Sonderzeichen gibt es allerdings Abkürzungen; es reicht " vor den Buchstaben zu setzten, so ergibt z. B. `"a` ä und `"s` ß².

Wie sieht nun ein Dokument aus? Ein kurzes Dokument ist in Abbildung 2.1 gezeigt. Zuerst steht die Präambel. Dort werden Grundeinstellungen vorgenommen. Mit `\documentclass` wird angegeben, was für ein Dokument vor-

¹Diese Einteilung stammt von mir!

² Mit Hilfe des Paketes `isolatin1` ist es auch möglich, die Umlaute und ß direkt einzugeben.

Zeichen	Bedeutung
%	Rest der Zeile ist Kommentar und wird nicht gedruckt
\	Ein Befehl beginnt
{ }	Optionen und Parameter eines Befehls
\$	Mathemodus beginnen oder beenden

Tabelle 2.1: Die wichtigsten Befehlszeichen

```
%Preamble:
\documentclass[a4paper,german]{article}
\usepackage[german]{babel}

\begin{document}
Hier steht mein erstes \LaTeX-Dokument!\
Das ist eine neue Zeile

Dies ist ein neuer Absatz!
\end{document}
```

Abbildung 2.1: Kurzes \LaTeX -Dokument

liegt, z. B. `slides`³ oder `book` oder wie hier `article`. In eckigen Klammern stehen die Optionen, hier für DIN A4-Papier und die deutschen Abkürzungen (Umlaute). Danach werden Pakete geladen. Es gibt sehr viele Pakete, einige werden noch vorgestellt, weitere werden z. B. in [3] beschrieben. Das Paket `babel` stellt Trenntabellen (z. B. für `ck` → `k-k`) zur Verfügung. Mit der Option `german` werden die deutschen Tabellen eingestellt. Es kann z. B. auch `french` gewählt werden.

Das eigentliche Dokument wird mit `\begin{document}` angefangen. `document` ist eine sogenannte Umgebung. Alles was nach `\end{document}` folgt, wird von \LaTeX nicht beachtet. In dieser Umgebung steht alles, was später gedruckt werden soll.

Da \LaTeX den Satz selbsttätig vornimmt, muß angegeben werden, wann ein neuer Absatz und damit eine neue Zeile begonnen werden soll. Dies geschieht entweder mit `\` (nur neue Zeile) oder durch Einfügen einer Leerzeile (neuer Absatz). Innerhalb eines Absatzes kümmert sich \LaTeX automatisch um den Umbruch und die Wortzwischenräume. Mehr dazu in [1].

2.3 Umgebungen

Die erste Umgebung `document` ist schon von oben bekannt. Es gibt noch viele weitere Umgebungen, die alle nach dem gleichen Schema funktionieren:

```
\begin{NameDerUmgebung}
Hier steht was gedruckt werden soll
```

³Zur einfachen Erstellung von Folien siehe Abschnitt 4.5

`\end{NameDerUmgebung}`

Wenn eine Umgebung beginnt, wird oft eine neue Zeile angefangen. Die wichtigsten Umgebungen sind in Tabelle 2.2 aufgelistet⁴.

Name	Bedeutung
itemize	Aufzählung soll stattfinden. Jeder Punkt wird mit <code>\item</code> eingeleitet.
enumerate	Ähnlich itemize, nur daß die Punkte numeriert werden.
verbatim	Alles in dieser Umgebung gesetzte wird ohne Veränderungen gedruckt. In diesem Dokument z.B. alle Quelltext-Beispiele. So läßt sich einfach z.B. C-Quelltext einbinden. Sollen ganze Dateien Eingebunden werden, so wird stattdessen <code>\verbatiminput{MeinQuelltext.c}</code> verwendet.
quote	Umgebung für Zitate.
table	In dieser Umgebung werden Tabellen gesetzt. Damit wird noch <i>nicht</i> die eigentliche Tabelle erzeugt. In eckigen Klammern kann die Position angegeben werden. Siehe dazu Abschnitt 4.2.
figure	Umgebung für Grafiken, ähnlich table.
tabular	Umgebung, in der die eigentliche Tabelle gesetzt wird. Siehe Abschnitt 4.3.
displaymath	Mathematik-Umgebung zum Formelsatz. Kann auch mit <code>\$</code> begrenzt werden, um innerhalb von normalem Fließtext Formeln zu setzen. Gut in [4] beschrieben!
eqnarray	Setzt eine nummerierte Formel. Soll die Nummer entfallen, so folgt ein <code>*</code> auf den Namen. Das Gleichheitszeichen wird mit <code>&=</code> zentriert. Sonst ähnlich displaymath.
thebibliography	Umgebung für Literaturangaben. Siehe [4] und Abschnitt 2.4.

Tabelle 2.2: Umgebungen

2.4 Ein langes Dokument

Jedes etwas längere Dokument braucht eine Gliederung. Ein (kurzes) langes Dokument, in dem vieles wichtiges bereits enthalten ist, ist in Abb. 2.2 dargestellt. Im ersten Abschnitt (in der Umgebung `titlepage`) wird die Titelseite erstellt. Mit `\maketitle` wird die Seite erzeugt. `\thispagestyle` regelt die Angabe von Seitenzahlen auf *dieser* Seite. `empty` sagt, daß keine gedruckt werden sollen. Als nächstes folgen jeweils auf einer eigenen Seite das Inhaltsverzeichnis, das Abbildungsverzeichnis und das Tabellenverzeichnis, wie in diesem Dokument auch. Sollten diese nicht gewünscht werden, werden die entsprechenden Zeilen einfach fortgelassen.

⁴Soll intensiv mit Formeln gearbeitet werden, empfiehlt sich der Einsatz von $\mathcal{A}\mathcal{M}\mathcal{S}\text{-I}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ (s. Abschnitt 6) anstatt der hier aufgelisteten Umgebungen.

```

\documentclass[a4paper,german,10pt,twoside]{book}
\usepackage{supertabular}
\usepackage[german]{babel}
\begin{document}
\begin{titlepage}
\title{Mein Titel}
\author {Nachname1, \\\ Vorname1 \and Nachname2, \\\ Vname2}
\date{\today}
\maketitle
\thispagestyle{empty}
\end{titlepage}
\newpage

\tableofcontents
\listoffigures
\listoftables

\chapter{Einleitung}\label{Einleitung}
\section{Mein erster Abschnitt}
Alle weiteren Kapitel beginnen auch mit chapter, section etc.

\subsection{Mein erster Unterabschnitt}
Mit Fu"snote\footnote{Meine Bemerkung}

\chapter{Ein weiteres Kapitel}
In Kapitel~\ref{Einleitung} schrieb ich

\section{Noch ein Unterkapitel}
Ich zitiere~\cite{marx}

\begin{thebibliography}{99}
\bibitem{marx}
Marx, \textbf{Kapital}, VEB-Osten
\end{thebibliography}
\end{document}

```

Abbildung 2.2: Ein langes Dokument

Das Dokument wird in Kapitel, Abschnitte, Unterabschnitte usw. gegliedert. Jeder neue „Abschnitt“ wird mit `\chapter` usw. begonnen, der Text in der Klammer erscheint im Inhaltsverzeichnis.

Fußnoten werden mit `\footnote` eingeleitet, in der Klammer folgt der Text der Fußnote. Hinter alle nummerierten Objekte (Abschnitte, Bilder, Tabellen, Gleichungen etc.) kann ein `label` gehängt werden. Damit kann jederzeit auf dieses Objekt verwiesen werden, indem einfach `\ref{Name}` geschrieben wird. Die Numerierung wird automatisch verwaltet. Die Tilde sorgt dafür, daß vor dem Verweis ein Leerzeichen eingefügt wird an dem kein Zeilenumbruch stattfinden darf. Analog zu `\ref{Name}` funktioniert `\cite` für Zitate.

Bis auf `chapter` gibt es alles auch im häufig benutzten `article`-Stil.

2.5 Was will T_EX mir sagen?

Wenn L^AT_EX beim Übersetzen des Dokumentes einen Fehler findet, dann gibt es eine oft kryptisch wirkende Fehlermeldung aus. Oft ist es sinnvoll, sich den Fehler grob zu merken (mit am wichtigsten ist die Zeilennummer) und einfach „q“ zu drücken. Dann versucht L^AT_EX das Dokument zuende zu übersetzen. Jetzt ist zu prüfen, ob in der Zeile oder kurz davor

- alle Umgebungsnamen richtig geschrieben wurden (z. B. nicht `enqarray`)
- die Matheumgebung begonnen oder geschlossen ist (fehlt ein `$`?)
- ein Befehl falsch geschrieben wurde (passiert oft bei Sonderzeichen)
- Klammerpaare richtig sind (jede Klammer muß irgendwo ihr Pendant haben)

Sollte dies nicht helfen, dann ist die fragliche Stelle so gut wie möglich zu isolieren und eine alternative Formulierung zu suchen (z. B. `\eqnarray*` statt `\displaymath`).

Daneben gibt L^AT_EX oft viele Warnungen (Warnings) aus. Betreffen diese Bezüge (Labels) oder Zitate und sind diese nicht falsch geschrieben, dann muß das Dokument erneut übersetzt werden. Dann sind diese Warnungen verschwunden. Warnungen wie `overfull hbox` bedeuten, daß L^AT_EX den Text nicht optimal setzen konnte. Oft befindet sich fälschlicherweise `\\` vor einem Abschnittwechsel. Durch Entfernen des Zeilentrenners verschwindet der Fehler dann. Manchmal ist es auch sinnvoll, den Text anders zu formulieren, insbesondere bei Schriftwechseln. Am besten wird dies am angezeigten Dokument sichtbar. Gelegentlich hilft es auch, manuelle Trennhilfen einzufügen [5].

Wenn etwas im Dokument fehlt, so ist auf `%`-Zeichen zu prüfen. Im Inhaltsverzeichnis tauchen die Einträge zudem erst nach dem *zweiten* Übersetzen des Dokumentes auf. Manchmal steht auch das `\end{document}` zu früh.

2.6 Logisches vs. physisches Layout

L^AT_EX trennt das logische Layout (d. h. Informationen wie Kapitel, Tabelle, Absatz) vom physischen Layout (d. h. Schriftgröße, Schriftart, Ränder, ...). Diese Trennung sollte auch innerhalb eigener Dokumente aufrechterhalten werden.

Um dies zu erreichen, sollte ein gewünschtes physisches Layout grundsätzlich über das Einbinden von Paketen oder Definitionen im Dokumentenkopf geschehen. Auf die direkte Verwendung von Befehlen innerhalb des Dokuments, die das physische Layout (z. B. die Schriftgröße) verändern, sollte verzichtet werden.

Oft sind beispielsweise spezielle Formatierungen gewünscht, in diesem Dokument sind dies

```
\newcommand*\strong[1]{\textbf{#1}}
\newcommand*\person[1]{\textsc{#1}}
\newcommand*\package[1]{\texttt{#1}}
\newcommand*\uri[1]{\%
  \chardef~^^>%
  \ttfamily #1%
}}
```

Jetzt kann mit `\strong` **betont**, mit `\person` **PERSONEN** ausgezeichnet und mit `\package` **Pakete** gekennzeichnet werden. Die Konstruktion bei `\uri` sorgt dafür, daß die Tilde innerhalb von Webadressen auch ausgegeben wird⁵. Siehe hierzu auch die Hinweise zur Verwendung von Farbe in Abschnitt 3.2.1.

Der Vorteil dieses Vorgehens besteht darin, daß zum einen sämtliche Personen leicht gefunden werden können (einfach nach `\person` suchen) und zum anderen, daß die Formatierung von Personennamen jederzeit geändert werden kann. Sollte keine Formatierung mehr gewünscht sein, dann reicht es, im Befehl die Formatierungsanweisung zu entfernen.

⁵Normalerweise dient die Tilde innerhalb von \LaTeX zur Kennzeichnung von Leerzeichen, bei denen nicht umgebrochen werden darf.

Kapitel 3

Graphik

3.1 \LaTeX und Graphik – ein Überblick

Wie oben erwähnt, wird Graphik von \LaTeX nicht direkt unterstützt. Die Graphikeinbindung geschieht mittels des Pakets `graphicx` bzw. `graphics` von DAVID CARLISLE. Der Unterschied zwischen den beiden Paketen besteht lediglich in der Aufrufsyntax der Befehle. Wenn möglich, sollten diese Pakete verwendet werden, eine Reihe von älteren Paketen¹ werden mittlerweile nur noch hierüber emuliert. Daher konzentriere ich mich in der folgenden Übersicht darauf. Eine komplette Anleitung des `graphicx`-Paketes ist in `epslatex.ps` und `grfguide.ps` in [6] sowie in [7] zu finden.

Im ersten Schritt muß die Graphik in ein für \LaTeX verwendbares Format gebracht werden. Normalerweise ist dies `postscript` oder `encapsulated postscript`, wobei letzteres vorzuziehen ist, da dort zusätzliche Informationen zum Einbetten der Graphik enthalten sind. Soll dagegen `pdf` erzeugt werden, so hat die Graphik im `pdf`, `tif` oder `png`-Format vorzuliegen. Es spricht nichts dagegen, beide Formate zu erzeugen, um so bei der Ausgabe flexibel zu bleiben. Hinweise zur Graphikkonvertierung finden sich in Abschnitt 5.6.

Für die eigentliche Graphikeinbindung muß in der Präambel `graphicx` eingebunden werden:

```
\usepackage{graphicx}
```

Die Einbindung der eigentliche Graphik geschieht jetzt mit

```
\begin{figure}[htbp]
\begin{center}
  \includegraphics[width=5 cm]{tux}
\end{center}
\caption{Der Linux-Pinguin.}\label{fig:tux}
\end{figure}
```

wie dies auch in Abbildung (3.1) gezeigt ist.

Die Figure-Umgebung sorgt dafür, daß das Bild im Text optimal plaziert wird (mehr Infos dazu in Abschnitt 4.2), das es ein Label² für spätere Referenzen

¹Eine Übersicht findet sich z. B. in [3]

²Referenzmarke



Abbildung 3.1: Eingebundenes png-Bild.

(siehe Abschnitt 4.4) und eine Bildunterschrift erhält. Die `center`-Umgebung zentriert das Bild.

Der hier eigentlich interessante Befehl ist `includegraphics`. In den geschweiften Klammern wird der Name des Bildes angegeben. Entfällt die Endung, dann versucht \LaTeX Bilder mit verschiedene Standardendungen (`eps`, `ps`, sowie komprimierte Fassungen davon) zu verwenden. Bei $\text{pdf}\text{\LaTeX}$ sind dies `pdf`, `tif` und `png`. In der `dvi/postscript`-Fassung dieses Dokumentes wird also `tux.eps` eingebunden, während dementsprechend bei der `pdf`-Fassung `tux.pdf` verwendet wird. Wird die Endung explizit angegeben, dann muß beim Wechsel von \LaTeX zu $\text{pdf}\text{\LaTeX}$ die Endung manuell angepaßt werden.

Im optionalen Argument in den eckigen Klammern können die in Tabelle 3.1 aufgelisteten Standardtransformationen auf das Bild angewendet werden, hier wird das Bild so skaliert, daß die Breite 5 cm beträgt. Ohne dieses Argument würde die Originalbreite verwendet (keine Skalierung).

Name	Bedeutung
<code>height</code>	Die Höhe, auf die die Graphik skaliert werden soll, inkl. Einheit.
<code>width</code>	Die Breite, auf die die Graphik skaliert werden soll, inkl. Einheit.
<code>scale</code>	Skalierfaktor, z. B. mit Verdopplung der Größe mit <code>scale=2</code> .
<code>angle</code>	Drehwinkel in Grad. Positive Werte entsprechen Drehungen gegen den Uhrzeigersinn.

Tabelle 3.1: Ein Auszug aus den Optionen des `includegraphics`-Befehls

3.2 Pakete rund um Graphik

3.2.1 Farbe

Auch die Ausgabe in Farbe ist mit L^AT_EX möglich. Es sollte jedoch mit Farbe im Text sparsam und einheitlich umgegangen werden. Für einen Vortrag auf Folien hatte ich z. B. folgende Definition:

```
\usepackage{color} % Option color für Farbe notwendig
%weitere Pakete
\newcommand{\FormelFarbe}{\color{blue}} % zentrale Farb-
\newcommand{\TabellFarbe}{\color{magenta}} % definition, damit
\newcommand{\TextFarbe}{\color{black}} % ist alles
% einheitlich
```

Jetzt kann z. B. folgende Konstruktion benutzt werden:

```
Der jetzt bewiesene Satz von Fermat
\begin{eqnarray*}
\FormelFarbe
\forall n > 2 \, , \, \nexists x, y, z \in \mathbb{N} :
x^n + y^n = z^n
\, , \, \text{\TextFarbe seit 300 Jahren}
\end{eqnarray*}
gilt nat"urlich nicht f"ur $\FormelFarbe n=2$:
```

Dies liefert – unter Verwendung der Pakete `amssymb` und `amsmath`, um `\text` (entspricht `\mbox` mit Verbesserungen) benutzen zu können, und für das Zeichen $\#$ – die (hier ohne Farbe wiedergegebene) Formel

$$\forall n > 2 \, \nexists x, y, z \in \mathbb{N} : x^n + y^n = z^n \text{ seit 300 Jahren}$$

Beim Arbeiten mit Farbe ist zu beachten, daß `xdvi` diese u. U. nicht darstellt, die zu druckende PostScript-Datei ist jedoch farbig. Farben können entweder direkt über ihren rot-grün-blau-Anteil oder besser als sogenannte *Named Colour* definiert werden, wie dies auch im Beispiel geschehen ist³. Die vordefinierten Farbnamen stehen in der Datei `color.pro` (bei mir im Verzeichnis `/usr/share/texmf/dvips/base`). Wichtige Named Colours sind Yellow, Magenta, Blue, Cyan, Green, Gray, Black, White. Um die Farben komplett wieder auszuschalten (z. B. für Probedrucke) reicht es, dem Paket `color` die Option `monochrom` anzugeben:

```
\usepackage[monochrom]{color}
```

Sollte die Farbausgabe nicht gelingen, ist es u. U. nötig, den Treiber als Option mitzuübergeben, d. h. z. B.

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

Dies reduziert natürlich die Portabilität.

Weitere Informationen (insbesondere zur Anpassung an spezielle Ausgabegeräte) finden sich in [6] und in [8].

³Leider ist dies nicht so portabel.

Kapitel 4

Tips und Tricks

4.1 Ich finde mein Zeichen nicht!

L^AT_EX verfügt über sehr viele Zeichen. Oft haben diese Zeichen kanonische Namen, die leicht erraten werden können: `\alpha` oder `\backslash` für α und `\.`. Als nächstes sollten die guten Tabellen in [4] verwendet werden. Dort sind alle Zeichen im mathematischen und im Text-Modus aufgelistet. Zum Teil wird dafür das Paket `latexsym` benötigt! Weitere (mathematische) Symbole sind im Paket `amssymb` definiert. Spezielle Graphiksymbole gibt es z. B. in den Paketen `Feynman`, `ChemTEX` und in einem Paket für Schaltkreise. Last but not least kann letztendlich jeder PostScript-Font eingebunden werden, so daß auch ausgefallene Schriften/Schnitte verwendet werden können.

Ein Typus Zeichen der (scheinbar) fehlt, ist die Gruppe der Mengensymbole, z. B. \mathbb{N} oder \mathbb{C} . Diese Zeichen heißen Mathematische-Fett-Tafel-Zeichen (mathematical blackboard bold) und werden z. B. im Paket `dsfont` definiert. Es bietet sich an, am Anfang, nachdem alle Pakete geladen sind, Anweisungen von der Form

```
\newcommand{\N}{\mathds{N}}
\newcommand{\Z}{\mathds{Z}}
```

stehen zu haben, so daß im Laufe des Textes auf diese Zeichen einfach mit `\Z` für \mathbb{Z} zugegriffen werden kann.

Eine alternative Implementation dieser Zeichen wird im Paket `bbm` vorgenommen. Dafür muß im Kopf des Dokuments

```
\usepackage{bbm}
\newcommand{\N}{\ensuremath{\mathbb{N}}}
\newcommand{\R}{\ensuremath{\mathbb{R}}}
\newcommand{\Eins}{\ensuremath{\mathbb{1}}}
```

stehen. Damit werden die Zeichen \mathbb{N} , \mathbb{C} , \mathbb{Z} , $\mathbb{1}$ usw. erzeugt.

4.2 Floats – und was getan werden kann

Die Umgebungen `figure` und `table` erzeugen sogenannte Gleitobjekte oder Floats. Alle Graphiken und alle Tabellen in diesem Dokument wurden so ge-

Buchstabe	Bedeutung
h	here – wenn möglich, an diese Stelle setzen
t	top – oben an die Seite setzen
b	bottom – unten an die Seite setzen
p	page – solange solche Objekte sammeln, bis es für eine Seite reicht, dann dort drucken

Tabelle 4.1: Mögliche Float-Parameter

setzt. Diese Objekte werden von \LaTeX auf der Seite selbsttätig positioniert, was nicht immer zum gewünschten Ergebnis führt.

Die einfachste Methode der Beeinflussung ist der Parameter `[x]`, der direkt nach dem Anfang der Umgebung folgt. `x` ist dabei einer von mehreren Parametern, s. hierzu Tab. 4.1. Eine oft sinnvolle Vergabe ist `[htbp]`, d. h. das Objekt soll sofort folgen, paßt dies nicht, dann auf der nächsten Seite oben, geht auch das nicht (was so gut wie nie der Fall ist), dann versuche es auf der nächsten Seite unten und ansonsten sammle die Objekte und gebe sie später aus.

4.2.1 Gleitobjekte im Fließtext

Für schmale Bilder oder Tabellen mag es sinnvoll sein, sie nicht im Seitenkopf oder -fuß auszugeben, sondern neben dem aktuellen Absatz, wie hier bereits demonstriert wird. Genau zu diesem Zweck stellt das Ergänzungspaket `floatflt` in Analogie zu den \LaTeX -Umgebungen `figure` und `table` die beiden Umgebungen `floatingfigure` und `floatingtable` zur Verfügung. Diese beiden Umgebungen arbeiten ähnlich wie `float` und `table`, erwarten aber keinen Positionierungsparameter wie `[htb]`, sondern eine Längenangabe, z. B. `{50mm}`, die die Breite der Abbildung bzw. Tabelle bestimmt.

```

Diese gleitende Abbildung wurde mit
\begin{floatingfigure}{58mm}
Diese gleitende ...
\caption{Eine gleitende ...}
\end{floatingfigure}

```

erstellt.

Abbildung 4.1: Eine gleitende Abbildung im aktuellen Absatz.

4.3 Tabellen

Tabellen stehen in der Umgebung `tabular`. Im Allgemeinen wird auch die Tabelle als fließendes Objekt realisiert und die gesamte Tabelle in die Umgebung `table` gesetzt. Eine solche Tabelle ist in Abbildung 4.2 dargestellt. Als Parameter werden der Tabelle die Ausrichtung der Einträge übergeben, also zentriert (`c`), linksbündig (`l`) oder rechtsbündig (`r`). Die Spaltenbreite wird automatisch ermittelt. Jede Zeile wird mit `\\` abgeschlossen, jede Spalte mit `&`. Die letzte Zeile wird mit dem letzten Eintrag beendet. Horizontale Linien werden mit `\hline` eingezeichnet, vertikale Linien im Ausrichtungsbereich (`l`). Die sich ergebende Tabelle ist in Tab. 4.2 wiedergegeben.

Das größte Problem sind mehrseitige Tabellen. Dafür gibt es die Umgebungen `supertabular` und `longtable`.


```

\begin{table}[htbp]
\begin{tabular}{|r|c|l|}
Wort & Arabisch & R"omisch \\
\hline
Eins & 1 & I \\
Zwei & 2 & II \\
Zehn & 10 & X
\end{tabular}
\end{table}

```

Abbildung 4.2: Eine normale Tabelle

Wort	Arabisch	Römisch
Eins	1	I
Zwei	2	II
Zehn	10	X

Tabelle 4.2: Ausgabe des Tabellenbeispiels

4.3.1 Lange Tabellen mit longtable

Die Verwendung der `longtable` Umgebung ist auch nicht schwieriger als die normaler Tabellen. Einfach das Paket `longtable` laden und `tabular` durch `longtable` ersetzen. Das ist schon das Grundgerüst. Dann muß nur noch angegeben werden, wie der Tabellenkopf auf der ersten und auf den weiteren Seiten und der Tabellenfuß auf der letzten und den vorherigen Seiten aussehen soll. Zusammengefaßt sieht die Struktur folgendermaßen aus:

```

\begin{longtable}{Spaltenform}
Tabellenkopf auf erster Seite
\endfirsthead
Tabellenkopf auf weiteren Seiten
\endhead
Tabellenfuß
\endfoot
Tabellenfuß auf letzter Seite
\endlastfoot
Tabellentext wie in einfachen Tabellen
\end{longtable}

```

Damit steht auch mehrseitigen Tabellen nichts mehr im Wege. Als Alternative wird nun noch die `supertabular` Umgebung vorgestellt. Der Anwender möge beide Umgebungen testen und nach seinem Geschmack entscheiden.

4.3.2 Lange Tabellen mit supertabular

Der Umstieg auf `supertabular` ist ebenfalls einfach. Das Paket `supertab` muß eingebunden und die fragliche Tabelle in `{supertabular}` statt in `{tabular}` eingeschlossen werden. Damit der Titel auf allen Seiten erscheint, wird vor Beginn der Tabelle die in Abb. 4.3 angeführten Zeilen eingegeben: Weitere Details stehen sehr gut in [3].

```

\tablehead{Die oberste Zeile}
\tabletail{\hline}
\tablecaption{Meine Bildunterschrift}

```

Abbildung 4.3: Supertabular-Zusatzbefehle

4.3.3 Professionelle Tabellen mit booktabs

\LaTeX verrichtet zwar schon gute Arbeit, Laien im Satzwesen professionelle Ausgabe zu ermöglichen, aber gerade im Bereich der Tabellen treten da noch viele Mängel auf. Man vergleiche nur einmal die folgenden beiden Tabellen:

Überschrift 1	Überschrift 2
Inhalt 1.1	Inhalt 2.1
Inhalt 1.2	Inhalt 2.2

Überschrift 1	Überschrift 2
Inhalt 1.1	Inhalt 2.1
Inhalt 1.2	Inhalt 2.2

Zuerst gibt es zwei goldene Regeln im Satzwesen, die man ruhig beherzigen sollte:

- Benutze nie senkrechte Linien!
- Benutze nie doppelte Linien!

Aber selbst unter Berücksichtigung dieser beiden Regeln ist das \LaTeX Ergebnis noch unbefriedigend:

Überschrift 1	Überschrift 2
Inhalt 1.1	Inhalt 2.1
Inhalt 1.2	Inhalt 2.2

Überschrift 1	Überschrift 2
Inhalt 1.1	Inhalt 2.1
Inhalt 1.2	Inhalt 2.2

Die Abstände zwischen den horizontalen Linien und dem Text ist bei der mit \LaTeX und `\hline` erzeugten Tabelle zu gering. Um dieses Defizit zu beheben, empfiehlt sich die Verwendung des Zusatzpakets `booktabs`, mit dem die jeweils rechten Tabellen erstellt wurden. Dieses Paket stellt drei Befehle `\toprule`, `\midrule` und `\bottomrule` bereit, der Quelltext für die obige Tabelle ist damit

```

\begin{tabular}{lc}
\toprule
\strong{"Überschrift 1}&\strong{"Überschrift 2}\\
\midrule
Inhalt 1.1&Inhalt 2.1\\
Inhalt 1.2&Inhalt 2.2\\
\bottomrule
\end{tabular}

```

4.4 Numerierung und Bezüge

Um sämtliche Numerierungen kümmert sich \LaTeX selbst. Verweise auf nummerierte Objekte (Bilder, Kapitel, Tabellen etc.) erfolgen *immer* nach dem gleichen

Schema. Hinter das zu numerierende Objekt (bei Umgebungen: nachdem sie geschlossen wurden, sonst direkt nach dem Befehl z. B. `\chapter{}`) wird der Befehl `\label{Titel}` angegeben. Der Titel muß im ganzen Dokument eindeutig sein, wobei Groß- und Kleinschreibung wichtig ist. Es empfiehlt sich, sinnvolle Namen zu wählen.

Später wird auf die Bezüge mit dem Befehl `\ref{Titel}` zugegriffen. Der Verweis kann auch vor dem Label stehen. Im allgemeinen empfiehlt es sich, einen Verweis in der Form

`in Kapitel\ref{KapFliegen} wird erklärt, wie ...`

zu schreiben, dann findet zwischen dem Wort „Kapitel“ und der eingefügten Nummer *kein* Zeilenumbruch statt. Wichtig ist zudem, daß alle Referenzen erst nach dem *zweiten* L^AT_EX-Lauf stimmen!

Manchmal werden Dokumente aus mehreren Texten zusammengesetzt, z. B. einem Text aus einer Textverarbeitung und einem aus L^AT_EX. Es ist dann ärgerlich, wenn die Kapitel-Nummern o. ä. nicht stimmen. Die aktuelle Nummerierungszahl kann jederzeit im Dokument mit

```
\setcounter{TheCounter}{Nummer}
```

geändert werden. Die Namen der Zähler sind kanonisch, also z. B. `chapter` für den Kapitelzähler.

Last but not least sind Fußnoten zu erwähnen. Sie werden einfach mit `\footnote{Der Fußnotentext}`¹ in den Text geschrieben und erscheinen auf der gleichen Seite unten, wenn die Fußnote noch paßt, sonst auf der nächsten Seite unten. Fußnoten in Tabellen sind schwierig und sollten vermieden werden (zumindestens am Anfang).

4.5 Folien

4.5.1 Echte Folien

Es gibt mittlerweile viele Pakete, die ein bequemes Erstellen von Folien und Dias (Slides) ermöglichen. Viele wissenschaftliche Institute haben Pakete, die das jeweilige Logo automatisch setzten. Ist kein solches Paket vorhanden, empfehle ich das `foils`-Paket das leider oft nicht mehr standardmäßig installiert und daher von [9] heruntergeladen werden muß.

Zuerst wird das Dokument wie gewohnt erstellt. Als Dokumentenklasse wird

```
\documentclass[20pt]{foils}
```

angegeben. 20 Punkt ist die Standard-Schriftgröße. Alternativ kann auch 17pt gewählt werden. Weitere Pakete, z. B. `dvips`, `draft`, ... können wie gewohnt ebenfalls eingebunden werden. Der zusätzliche Arbeitsbedarf beschränkt sich auf das Einstellen einiger weniger Parameter, der komplette Satz wird automatisch folienangepaßt durchgeführt.

Nach dem `\begin{document}` können die einzelnen Statusausgaben eingestellt werden. Jede Einstellung bleibt für alle Folgeseiten erhalten, bis der gleiche Befehl diese Einstellung ändert. Die Befehle sind in Tabelle 4.3 aufgeführt. Weitere Informationen sind der Dokumentation des Pakets zu entnehmen.

¹Der Fußnotentext

Befehl	Auswirkung
<code>\foilhead{Text}</code>	Neue Seite mit Überschrift „Text“ beginnen
<code>\rotatefoilhead{Text}</code>	Neue Seite um 90° gedreht mit Überschrift „Text“ beginnen
<code>\righthead{Text}</code>	Text oben rechts auf „Text“ setzen
<code>\lefthead{Text}</code>	Text oben links auf „Text“ setzen
<code>\rightfooter{Text}</code>	Text unten rechts auf „Text“ setzen
<code>\Restriction{Text}</code>	Copyright-Text auf „Text“ setzen
<code>\MyLogo{Text}</code>	Logo-Text auf „Text“ setzen

Tabelle 4.3: Foil \TeX -Steuer-Befehle

4.5.2 Präsentationen mit dem Computer

Um Präsentationen mit dem Rechner zu erstellen, wird nicht unbedingt der „Kraftpunkt“ benötigt. Als Präsentationsprogramm wird statt dessen der Acrobat-Reader [10] verwendet, d. h. die Präsentation ist plattformunabhängig, erfordert dafür aber die Erzeugung einer pdf-Datei. Bei `ppower4` [11] wird mit `pdflatex` (s. Abschnitt 9) gearbeitet, während `texpower` [12] die postscript-Datei zum Schluß in pdf wandelt. Ich beschränke mich hier auf `ppower`.

Als Grundlage dient auch hier wieder das `foils`-Paket (s. Abschnitt 4.5.1). Die mit `pdflatex` erzeugte pdf-Datei wird mit einem java-Programm (das von [11] heruntergeladen werden muß) noch nachbearbeitet. Im folgenden der Ablauf zur Erstellung der Präsentation:

1. Erstellen der Folien (s. Abschnitt 4.5.1)
2. Einbinden des Pakets `ppower4` durch

```
\usepackage{pause}      % eigentliche Funktionalität
\usepackage{background} % für Hintergründe
\usepackage{pp4slide}   % verbessert Layout
```

möglichst gegen Ende der Paketdefinitionen. Ich empfehle zudem die Verwendung des Pakets `hyperref` (s. Abschnitt 9.4.1).

3. An jeder Stelle, an der die Präsentation warten soll, wird der Befehl `\pause` eingefügt
4. Die Art des Überblendeffektes wird mit

```
\hypersetup{pdfpagetransition={Blinds /Dm /H}}
```

eingestellt, sie gilt immer für eine logische² Seite. Weiter mögliche Effekte sind in Dateien, die von der Pakethomepage [11] aus verlinkt sind, erläutert.

²Der Acrobat-Reader realisiert Überblendungen, in dem er jede Änderung als neue Seite speichert. Für den Betrachter sieht es aber so aus, als ob die Seite Schrittweise aufgebaut wird.

5. Der Hintergrund kann als Farbverlauf realisiert werden. Dazu werden im Dokument z.B. folgende Befehle verwendet:

```
% eigene Farbe definieren
\definecolor{bgblue}{rgb}{0.04,0.39,0.53}
% vertikal verlaufen lassen
\vpagcolor{bgblue}
```

Als Farbe kann auch jede \LaTeX Standardfarbe verwendet werden (s. Abschnitt 3.2.1). Für horizontale Verläufe wird der Befehl `\hpagcolor` verwendet. Soll von einer Farbe in eine andere verlaufen werden, wird als zweites Argument die zweite Farbe übergeben, z.B.

```
\vpagcolor{blue}{red}
```

Für einfarbige Hintergründe dient der Befehl `\pagcolor{farbe}`.

6. Das Dokument wird mit `pdflatex` übersetzt (u. U. mehrfach um alle Quer-
verweise aufzulösen).
7. Mit der im Paket beigefügten Skript- bzw. Batchdatei wird die eigentliche
Präsentation aus der pdf-Datei erstellt, also z. B.

```
ppower4 meindokument.pdf meineshow.pdf
```


Kapitel 5

Editoren und andere Hilfsmittel

5.1 \LaTeX und Unix (Linux)

\LaTeX wird von verschiedenen Personen zusammengestellt und gewartet. Innerhalb einer Plattform sind die (kleinen) Unterschiede sehr selten bemerkbar, sie manifestieren sich lediglich in leicht verschiedenen Verzeichnisnamen. Im folgenden beschreibe ich alles auf Basis der $\text{te}\LaTeX$ -Konfiguration, für andere Zusammenstellungen (z. B. $\text{NT}\LaTeX$) können Details anders sein.

Am Anfang sollte auf jeden Fall als root `texconfig` aufgerufen werden. Dieses Programm ermöglicht es, einige Voreinstellungen (z. B. Druckerauflösung) vorzunehmen, so daß später der Drucker optimal ausgenutzt wird. Diese Einstellungen sind z. B. für `xdvi` und `dvips` wichtig (beide werden gleich erklärt) und unabhängig von globalen Druckereinstellungen für das Drucksystem (z. B. `lpr` oder `cups`).

Als nächstes steht die Wahl des Editors an. Es gibt eine schier unüberschaubare Anzahl an Editoren, z. B. `vi(m)`, `emacs`, `axe`, `joe` u. v. m. Von allen Editoren ist `emacs` sicherlich der umfangreichste, aber auch der gewöhnungsbedürftigste Editor. Jeder sollte den für ihn/sie angenehmsten Editor verwenden.

Da Linux über echtes¹ Multitasking verfügt, bietet es sich an, das auszunutzen. Am Beispiel einer Umgebung basierend auf X11 mit dem Fenstermanager `fvwm2` – für andere Umgebungen sind die Tastaturbelegungen ggf. anders – soll hier kurz eine praktische Arbeitsumgebung geschildert werden.

Auf einem virtuellen Bildschirm wird ein Terminalfenster, z. B. `xterm`, geöffnet. Oft ist beim Start von X bereits eines offen. Dieses `xterm` dient der Verwaltung. Heißt der Text z. B. `MeinText.tex`, so lautet die Eingabe in diesem `xterm`:

```
latex MeinText.tex && xdvi MeinText.dvi &
```

Während der Text das erste Mal übersetzt wird, wird auf einem zweiten virtuellen Bildschirm der gewünschte Editor mit dem Quelltext geöffnet. Ist der Quelltext geändert, wird entweder aus dem Editor heraus das Dokument neu übersetzt (so z. B. beim `emacs`) oder mit `Alt-F11` bzw. `Alt-F12` auf das `xterm`-Fenster gewechselt und dort mit

¹Damit ist präemptives Multitasking gemeint.

```
latex MeinText.tex
```

das Dokument neu übersetzt. Bei späteren Läufen reicht dann bereits ein Druck auf die Bild ↑-Taste. Sobald das Dokument (fehlerfrei) übersetzt worden ist, wird wieder mit **Alt-F11** bzw. **Alt-F12** auf das **xdvi**-Fenster gewechselt. **xdvi** lädt beim Aktivieren immer die neuste (aktuelle) Version des Textes, so daß es im Hintergrund immer laufen sollte. Der Programm-Wechsel kann alternativ auch mittels der Maus erfolgen.

Da der Vorgang immer der gleiche ist, bietet es sich an, ihn mittels eines Shell-Skripts zu automatisieren. Das Skript in Abbildung 14 erfüllt diese Aufgabe.

```
#!/bin/bash
#Diese Datei uebersetzt immer, wenn sich in der Quelldatei etwas
#getan hat
#
#xdvi Haupt.dvi &
xdvi $1.dvi
tmp2=$(ls -l $1.tex)
while true; do
  if [ "$tmp1" != "$tmp2" ]; then
    tmp2=$(ls -l $1.tex)
    echo "Ich uebersetzte jetzt"
  #   latex Haupt.tex
  latex $1.tex
  echo -e "\a"
  fi
  sleep 3
  tmp1=$(ls -l $1.tex)
done
```

Abbildung 5.1: bash-Skript zur \LaTeX -Automatisierung

Das Skript wird unter z. B. **mytex** abgespeichert. In Zukunft reicht es, einmal **mytex MeinText** (ohne **.tex**-Endung!) einzugeben, dann wird das Dokument jedesmal neu übersetzt, wenn im Editor gespeichert wird. Damit der Benutzer informiert ist, daß wieder eine aktuelle Version seines Dokumentes im **xdvi** vorliegt, ertönt nach jedem Übersetzungslauf ein Piepser. Der Benutzer kann jetzt wie oben beschrieben in das **xdvi**-Fenster wechseln und sein Dokument betrachten. Wichtig ist, daß immer eine **dvi**-Datei vorliegt. Ansonsten sollte vor Zeile 5 noch die Zeile

```
latex $1.tex
```

eingefügt werden, damit wird das Dokument auch beim ersten Skript-Durchlauf neu übersetzt.

Sind Fehler beim Übersetzten aufgetreten, so sollte der \LaTeX -Lauf mit **q** beendet werden und der Fehler behoben werden. Im **xdvi** erscheint bis dahin eine halbfertige Version. Das Skript wird beendet, indem es mit **Strg-C** abgebrochen wird.

Eine interessante Variante stellen die auskommentierten Zeilen dar. Kommentarzeilen beginnen in Shellskripten immer mit einem **#** am Anfang der Zeile. Bei

großen Projekten bietet es sich an, eine zentrale Datei anzulegen (`Haupt.tex`) und die einzelnen Kapitel/Abschnitte in eigenen Dateien zu speichern. Im Skript werden dazu die zwei Zeilen, die `Haupt.tex` enthalten, entkommentiert (`#`-entfernen) und die jeweils folgende Zeilen auskommentiert (`#` am Zeilenanfang einfügen). Das Skript wird wie gewohnt verwendet, nur übersetzt es jetzt die richtige Hauptdatei.

Auch diese Skript-Variante ist noch nicht perfekt. Als erstes ist die Haupt-Datei fest verdrahtet. Dies kann geändert werden, indem die Zeichenfolge `Haupt` durch `$2` ersetzt wird. Dann wird das Skript mit zwei Parametern gestartet:

```
mytex MeinText Haupt
```

Ferner fehlt die Fehlerbehandlung komplett, auch sind weitere Ergänzungen denkbar. Das ursprüngliche Skript sowie gute Skript-Programmierhinweise sind in [13] zu finden.

Eine Alternative zu diesem direkten Vorgehen bieten sogenannte \LaTeX -Arbeitsumgebungen, z. B. `xtem` [14], `ts` [15] und `ktexshell` [16]. Sie haben oft viele \TeX -Hilfsfunktionen und sollten durchaus ausprobiert werden. Als vim-Shell-Fan gehe ich hier nicht darauf ein.

Last but not least sind noch ein paar Worte über `dvips`. Dieses Programm wandelt `dvi`-Dateien in `PostScript`-Dateien um, die dann unter (fast) allen Unixen problemlos gedruckt werden können. Mit ihm können einige Seiten, Seitenbereiche, alle geraden/ungeraden Seiten oder auch das ganze Dokument umgewandelt werden. Im Normalfall (ganzes Dokument) wird das Dokument einfach mit

```
dvips MeinText.dvi
```

umgewandelt. Ist \LaTeX richtig konfiguriert (s. o. `texconfig`), dann wird das Dokument umgewandelt und gedruckt. Ich empfehle dringend, mittels

```
man dvips
```

die Manual-Pages zu lesen und die Optionen durchzugehen!

5.2 \LaTeX und DOS

Unter DOS ist das Paket `Em \TeX` sehr verbreitet. Wird direkt unter DOS ge \TeX ed, so sollte ein möglichst einfacher Editor, z. B. `edit`, der bei DOS dabei ist, verwendet werden, um Arbeitsspeicher zu sparen.

Auch bei DOS empfiehlt es sich, eine Start-Batch-Datei zu erstellen. Eine beispielhafte Batch-Datei ist in Abb 5.2 skizziert. Diese Batchdatei soll lediglich als grober Anhaltspunkt verstanden werden. Es ist ferner sinnvoll, die Funktionstasten mit bestimmten Programmen zu belegen, z. B. `F11` für den `dvi`-Darsteller, `F12` für \LaTeX usw. Weitere Tips und Tricks nehme ich gerne auf.

5.3 \LaTeX und Windows

Für Windows² sind `Em \TeX` `MiK \TeX` und `fp \TeX` ³ die bekanntesten \TeX -Pakete. Unter der Adresse

²Vielen Dank für diesen Abschnitt an CHRISTIAN JENTSCH.

³Hierbei handelt es sich um eine Portierung von `te \TeX` auf Windows.

```

@echo off
rem Zuerst die Umgebungsvariablen setzten (mTeX)
rem
rem   DVIDRVINPUT      Suchpfad fuer DVI Dateien.
rem   DVIDRVFONTS      Suchpfad fuer FLI Dateien.
rem   DVIDRVGRAPH      Suchpfad fuer graphic Dateien.
rem
rem Die Pfade sind natuerlich anzupassen!
set dvidrvinput=E:\DATEN\TEXTE\tex
set dvidrvfonts=D:\emtex\texfonts
set dvidrvgraph=c:\mytex;c:\emtex\doc
rem
rem doskey starten, zum bequemeren arbeiten
DOSKEY
rem
rem Laufwerk & Verzeichnis setzten
E:
cd \daten\texte\tex
rem das wars ...

```

Abbildung 5.2: Beispielhafte DOS-Batch-Datei für \LaTeX

<http://infaut.et.uni-magdeburg.de/~werner/winemtex.htm> findet sich ein Paket von EmTeX^4 (Version 3.14159), das zusätzlich einen Editor, einen dvi- sowie einen PostScript-Viewer enthält. Von den 16 Disketten wird der gesamte Programmumfang unter Windows automatisch durch Starten der Datei `texinst.exe` installiert. Je nach Konfiguration wird dafür ein Platz von 10–45 MB auf der Festplatte benötigt. Abgefragt wird auch, ob für Windows '95/'98 die 32 Bit-Version oder für Windows 3.11 die 16 Bit-Variante genutzt werden soll.

Zum Lieferumfang gehört als Benutzeroberfläche der Editor WinEdit 96V von STEVE SCHAUER. Der große Vorteil besteht darin, daß für diesen Editor Menüfunktionen mitinstalliert werden, die direkt auf EmTeX zugeschnitten sind (z. B. sind schon verschiedene Makros vorhanden, weitere lassen sich aufzeichnen). Weiterhin kann direkt von der Oberfläche heraus kompiliert bzw. die Viewer oder TEXCAD gestartet werden. Als weitere Hilfsprogramme werden der dvi-Viewer `DviWin 2.9` von HIPPOCRATES SENDOUKAS in der Version von Ende 1994 sowie der PostScript-Viewer `GhostView` in der Version 2.1 von Ende 1996 mitgeliefert. Sofern kein postscriptfähiger Drucker vorhanden ist, kann prinzipiell auch aus dem dvi-Viewer gedruckt werden. Jedoch muß in diesem Fall etwas mit den Auflösungen herumgespielt werden, um das optimale Schriftbild zu erreichen⁵. Ein kleines Problem ergibt sich noch, wenn das Paket `babel` verwendet wird: Es muß zunächst kompiliert (\rightarrow `babel.sty`) und dann in eines der `Style`-Verzeichnisse kopiert werden.

Insgesamt liegt mit diesem Programmpaket jedoch eine vollständige und vor allen Dingen sofort nutzbare TeX -Version vor (d. h. keine Pfade müssen in Win-

⁴Für Erfahrungsberichte/Tips auch und insbesondere mit fpTeX bin ich dankbar.

⁵Evtl. muß im Editor unter dem Menüpunkt „Hilfsprogramme – Edit Utility Menu“ die Variable „printertype“ geändert werden.

dows-Systemdateien ergänzt werden o. ä.), die auf jeden Fall weiterzuempfehlen ist.

Das zweite Paket MiK_TE_X (Version 1.10b) ist z. B. auf [17] enthalten oder kann von [18] heruntergeladen werden. Zum Umfang gehört ein vollständiges T_EX-System, was aber ohne Windows-Nutzeroberfläche am besten über Batchdateien gesteuert wird. Der zu MiK_TE_X gehörende dvi-Viewer YAP, Version 0.93c, von CHRISTIAN SCHENK ist einfacher gestaltet als DviWin, allerdings durch weniger Optionen auch weniger verwirrend. Besonders nützlich ist, daß der Mauszeiger zu einer kleinen Hand wird, mit der der Inhalt der gerade angezeigten Seite innerhalb des YAP-Fensters verschoben werden kann, ohne dazu die Bildlaufleisten zu benutzen. Für die Installation von MiK_TE_X werden ebenfalls ca. 40 MB auf der Festplatte benötigt.

Als geeignete Editoroberfläche kann z. B. das Programm WinTex95, Version 2.02 von MICHAEL MÜCKE dienen. Dieser Editor ist ebenfalls auf der oben genannten CD vorhanden bzw. unter [19] erhältlich. Wie der Name schon andeutet, eignet sich dieser Editor nur für Windows 95. Ein Nachteil ist allerdings, daß zwar viele Sonderzeichen per Mausklick in den T_EX-Quelltext eingefügt werden können, jedoch der Editor nicht für eine bestimmte T_EX-Version ausgelegt ist, d. h. Befehle zum Übersetzen usw. müssen zunächst selbst in den Optionen definiert werden.

Ohne eine Wertung der Vor- und Nachteile von EmT_EX und MiK_TE_X vorzunehmen zu wollen, würde ich T_EX-Anfängern eher zum EmT_EX-Paket raten, weil gleich nach der Installation ein vollständig funktionsfähiges System vorliegt.

5.4 Emacs und der T_EX-Modus

Einer der leistungsfähigsten Editoren ist sicher der emacs. Er verfügt nicht nur über eine immense Flexibilität (Programmierbarkeit), sondern enthält bereits vorkonfigurierte Umgebungen für verschiedene Dokumente. Der emacs ist übrigens auch für Windows verfügbar (s. [20]).

Ich möchte hier nicht auf die emacs-Bedienung⁶ im Detail eingehen, fast jedes Buch über Unix/Linux enthält eine Einführung und Referenz, so z. B. [21], [13], [22] u. v. m., sondern lediglich einige Eigenschaften des T_EX/L^AT_EX-Modus erwähnen.

Der L^AT_EX-Modus wird automatisch geladen, wenn die Datei die Endung `.tex` besitzt. Ist die Endung nicht klein geschrieben, z. B. `.Tex` so kann es sein, daß stattdessen der T_EX-Modus geladen wird, in dem einige der folgenden Dinge nicht gelten. In manchen Konfigurationen wird das AucT_EX-Paket installiert, welches einen eigenen, anders gearteten L^AT_EX-Modus bereitstellt, so unterstützt AucT_EX z. B. die Einfärbung der Befehlswörter. Mit AucT_EX habe ich wenig Erfahrung und werde deshalb hier nicht darauf eingehen. Anmerkungen dazu sind natürlich erwünscht.

Einige wichtige Tastenbefehle sind in Tabelle 5.1 aufgelistet. Dabei steht **Strg** für die Strg (auf englischen Tastaturen Control)-Taste und **Meta** für die Meta/Esc-Taste, die oft auch auf die Alt-Taste gelegt wird. Ein Bindestrich bedeutet, daß zwei Tasten gleichzeitig gedrückt werden müssen. Unter X verfügt der emacs zudem über eine Menüleiste, die es ermöglicht, unbekannte Befehle

⁶Auch auf Unterschiede zum Xemacs verweise ich auf die Literatur

Tasten	Bedeutung
Strg-a	Anfang der Zeile
Strg-e	Ende der Zeile
Strg-x f Name	Datei laden
Strg-x s	Datei speichern
Strg-Leertaste	Markierung beginnen
Strg-w	markiertes löschen
Strg-y	gelöschtes einfügen
Strg-c f	L ^A T _E X-Dokument übersetzen
Strg-x 1	Aktuelles Teilfenster zum Vollfenster vergrößern

Tabelle 5.1: Wichtige emacs-Tastenkombinationen

per Maus auszuführen. Oft hilft es auch, einfach `Meta-x` zu tippen. Dann kann ein Befehl in Langform eingegeben werden, z. B.

`Meta-x search-forward`

um vorwärts zu suchen. Mit der Tabulatortaste wird der Befehl vervollständigt bzw. noch alle möglichen angezeigt.

Wie oben schon erwähnt, ist der emacs extrem flexibel. Glücklicherweise ist vieles schon sinnvoll vorkonfiguriert. Für den deutschsprachigen L^AT_EX-Benutzer ist allerdings die Belegung der "-"Taste sehr unglücklich. Um diese Taste sinnvoll zu nutzen, sind die in Abbildung 5.3 angegebenen Zeilen in der lokalen (im home-Verzeichnis befindlichen) `.emacs`-Datei einzufügen. Sollte die Datei noch nicht existieren, kann sie unbedenklich angelegt werden. In dieser Datei kann

```
(add-hook 'tex-mode-hook
  (function (lambda ()
    (setq tex-open-quote 34)
    (setq tex-close-quote 34)
  )))
```

Abbildung 5.3: Ergänzungen für `.emacs`

für jeden emacs-Modus die Einstellungen *verändert* werden. Ist z. B. AucT_EX installiert, kann ein Blick in die Datei nicht schaden (die Zeilen sind leicht zu finden und auszukommentieren). Eine gute Einführung die die Programmierung des emacs ist in [13] gegeben, viele Hinweise, Beispiel-Skripte u. ä. sollen auf

<ftp://archive.cis.ohio-state.edu/pub/gnu/emacs/elisp-archive>

vorhanden sein⁷, allerdings kann ich dazu nichts schreiben.

⁷Linux-Magazin 3/97, Seite 53, hier steht auch noch einiges zu elisp, der Programmiersprache des emacs.

5.5 Mein PostScript-Bild ist nicht ok

Der unter Unix gebräuchlichste Weg der Graphik-Beschreibung ist sicherlich (encapsulated) PostScript. Fast alle Programme unter Unix können dieses Format schreiben. Gelegentlich ist das PostScript-Bild nicht ganz wie gewünscht. Anstatt aufwendig im Erzeugungsprogramm das Bild zu manipulieren, lohnt es sich u. U., mit dem Programm `pstoedit` [23] das PostScript-Bild in das (x)fig-Format zu konvertieren, um es dann mit `xfig` [24] nachzubearbeiten.

5.6 Ein paar Worte zur Graphikkonvertierung

Gerade im Zusammenhang mit pdf_TE_Xmüssen oft Graphiken sowohl im eps als auch im png/tiff-Format vorliegen. Wenn die erzeugende Anwendung nicht beide Formate schreiben kann, muß das eine Format in das andere Format konvertiert werden. Im folgenden eine kurze Liste von Programmen⁸, die hierzu prinzipiell geeignet sind; im konkreten Anwendungsfall ist jeweils das optimale zu ermitteln. Auch der „Umweg“ über xfig (siehe Abschnitt 5.5) kann im Einzelfall geeignet sein.

convert Teil der Image Magick Tool Suite [25], kann sehr viele Formate ineinander konvertieren und darstellen

epstopdf Teil der pdf_TE_X-Programmsammlung

5.7 Hinweise zum Suchen von Paketen, Optionen etc.

Für konkrete Fragen zu einzelnen Paketen sind die erste Anlaufstelle die in der L^AT_EX-Distribution enthaltenen Beschreibungen [26], die sich sehr übersichtlich mit dem Programm `texdoctk` [27] darstellen lassen.

Weitere Informationen zu L^AT_EX sind auf <http://www.dante.de> zu finden. Darunter sind eine große Auswahl an Paketen (inkl. Verweise auf die Heimatseiten der Autoren) sowie eine ausführliche FAQ⁹.

Aus der Vielzahl von Büchern sind aus meiner Sicht der L^AT_EX-Begleiter von Goossens et. al [3] sowie die L^AT_EX-Bücher von Kopka [28, 29, 30] zu empfehlen. Bei sehr speziellen Problemen, die durch die obigen Medien nicht abgedeckt werden, lohnt es sich, diese in der Newsgruppe `de.comp.text.tex` zu posten. Dort sind eine Reihe von Leuten sehr bemüht zu helfen. Um die Qualität der Gruppe zu erhalten, sollte aber auf jeden Fall die Archive und die oben genannten anderen Quellen vorab konsultiert werden.

⁸Die hier aufgelisteten Programme beschränken sich auf Unix/Linux-Programme; Hinweise für andere Plattformen werden gerne entgegengenommen

⁹Frequently asked questions – häufig gestellte Fragen.

Kapitel 6

$\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX

Obwohl \LaTeX im Vergleich zu nicht \TeX -basierter Software ausgezeichnete Arbeit beim Satz mathematischer Formeln liefert, fand die AMERICAN MATHEMATICAL SOCIETY ($\mathcal{A}\mathcal{M}\mathcal{S}$) die Ergebnisse ungenügend und hat eine Erweiterung namens $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX geschaffen, um die Schwächen von \LaTeX auszumerzen sowie zusätzliche Befehle bereitzustellen.

Das $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX Paket heißt `amsmath` und wird somit mittels

```
\usepackage{amsmath}
```

eingebunden. Die möglichen Optionen für dieses Paket sind in Tab. 6.1 angegeben.

Option	Beschreibung
<code>centertags</code>	(Standard) Gleichungsnummern vertikal zentriert bei mehrzeiligen Formeln
<code>tbtags</code>	Gleichungsnummern an der letzten (bzw. ersten) Zeile, wenn die Gleichungsnummern auf der rechten (bzw. linken) Seite stehen.
<code>sumlimits</code>	(Standard) Indizes an Summationszeichen \sum unter- bzw. oberhalb dieser. Betrifft auch Zeichen wie Produktzeichen \prod u. ä.
<code>nosumlimits</code>	Indizes an Summationszeichen u. ä. neben den Summationszeichen
<code>intlimits</code>	Wie <code>sumlimits</code> , nur für Integrale
<code>nointlimits</code>	(Standard) Wie <code>nosumlimits</code> , nur für Integrale
<code>namelimits</code>	(Standard) Wie <code>sumlimits</code> , nur für bestimmte Operatornamen wie z. B. <code>lim</code>
<code>nonamelimits</code>	Wie <code>nosumlimits</code> , nur für bestimmte Operatornamen
<code>leqno</code>	Gleichungsnummern links
<code>reqno</code>	Gleichungsnummern rechts
<code>fleqn</code>	Gleichung linksbündig mit festem Einzug statt zentriert

Tabelle 6.1: Optionen des Pakets `amsmath`

Weiterhin von nutzen ist das Paket `amssymb`, das gegenüber `latexsym` zusätzliche Symbole bereitstellt.

Soll $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ verwendet werden, so ist von einer Verwendung der Umgebungen `displaymath` und `eqnarray` bzw. `eqnarray*` *dringend abzuraten!*

6.1 Gleichungsumgebungen

$\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ stellt deutlich mehr Gleichungsumgebungen für die verschiedensten Zwecke bereit als $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ selbst. Mit einer Ausnahme stehen für alle Umgebungen gesternte und ungesternete Versionen zur Verfügung, die ungesterten für Formeln mit Gleichungsnummern und die gesterten für Formeln ohne Gleichungsnummern.

6.1.1 Einzelne Gleichungen

`equation*`

Nach dieser einfach zu merkenden Regel wird die $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ -Umgebung `displaymath` durch die Umgebung `equation*` abgelöst, einfache Formeln ohne Nummern sind also durch

```
\begin{equation*}
...
\end{equation*}
```

zu begrenzen.

`multline`

Soll sich eine Formel über mehrere Zeilen erstrecken, ohne daß dabei eine besondere Ausrichtung der Zeilen zueinander notwendig ist, dann ist hierfür die Umgebung `multline` bzw. `multline*` geeignet. Die einzelnen Zeilen werden dabei durch `\\` voneinander getrennt.

```
\begin{multline*}
a+b+c+d+e+f\\
+i+j+k+l+m+n
\end{multline*}
```

$$a + b + c + d + e + f \\ + i + j + k + l + m + n$$

`split`

Ist hingegen die Ausrichtung wichtig, so ist die `split`-Umgebung die richtige Wahl. Im Gegensatz zu `multline` muß `split` in eine `equation`- oder `equation*`-Umgebung eingebettet werden. Die einzelnen Zeilen werden durch `\\` getrennt und an dem Zeichen `&` ausgerichtet. Tritt mehr als ein `&` pro Zeile auf, so wird ein Fehler ausgegeben.

```
\begin{equation*}
\begin{split}
a&=b+c-d\\
&\quad+e-f\\
&=g+h\\
&=i
\end{split}
\end{equation*}
```

$$a = b + c - d \\ \quad + e - f \\ = g + h \\ = i$$

6.1.2 Gleichungsgruppen

Für Gleichungsgruppen stehen ebenfalls mehrere Umgebungen zur Verfügung. Ohne spezielle Ausrichtung sollte die `gather`- bzw. `gather*`-Umgebung verwendet werden. Wie gehabt werden die Zeilen durch `\` getrennt.

```
\begin{gather*}
a_1=b_1+c_1\\
a_2=b_2+c_2-d_2+e_2
\end{gather*}
```

$$a_1 = b_1 + c_1$$

$$a_2 = b_2 + c_2 - d_2 + e_2$$

Sollen die Gleichungsgruppen dagegen ausgerichtet sein, so ist im einfachsten Fall die `align`- bzw. `align*`-Umgebung zu verwenden. Zeilen werden wie gehabt durch `\` getrennt, die Ausrichtung erfolgt an jedem ungeraden Auftreten des `&`-Zeichens. An jedem geraden Auftreten des `&` wird Leerraum in benötigter Breite eingefügt.

```
\begin{align*}
a_{11}&=b_{11}&
a_{12}&=b_{12}\\
a_{21}&=b_{21}&
a_{22}&=b_{22}+c_{22}
\end{align*}
```

$$a_{11} = b_{11} \quad a_{12} = b_{12}$$

$$a_{21} = b_{21} \quad a_{22} = b_{22} + c_{22}$$

6.1.3 Gleichungsteile

Auch für mehrzeilige Teile von Gleichungen gibt es mehrere Umgebungen, die allesamt nur in einer mathematischen Umgebung auftreten können. Ohne spezielle Ausrichtung kommt die `gathered`-Umgebung zum Zuge, das Pendant zur `gather`-Umgebung.

Mit Ausrichtung sollte die `aligned`-Umgebung verwendet werden, das Gegenstück zur `align`-Umgebung.

Für Fallunterscheidungen steht die `cases`-Umgebung zur Verfügung. Die beiden Felder – mathematischer Ausdruck und Bedingung – sind durch ein `&` voneinander getrennt, die einzelnen Fälle durch `\`.

```
\begin{equation*}
\bigl\lvert x \bigr\rvert =
\begin{cases}
x & \text{if } x \ge 0 \\
-x & \text{if } x < 0
\end{cases}
\end{equation*}
```

$$|x| = \begin{cases} x & \text{if } x \geq 0 \\ -x & \text{if } x < 0 \end{cases}$$

6.1.4 Weiteres

Manchmal sollen Gleichungen als Zusammenhängend gekennzeichnet werden, indem die Gleichungsnummer gleichbleibt, dafür aber ein angehängter kleiner Buchstabe, die Gleichungen durchnummeriert. Dies kann mit der `subequations`-Umgebung erreicht werden.

Soll in einer ungestrichelten Gleichungsumgebung für eine Zeile eine Gleichungsnummer unterbunden werden, so ist vor dem Zeilentrenner `\` der Befehl `\notag`

subequations

\notag

einzufragen.

`\displaybreak`

Standardmäßig dürfen bei der Verwendung von $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ keine Seitenumbrüche innerhalb von mathematischen Formeln auftreten. Bei Bedarf kann manuell vor einem Zeilenumbruch `\\` der Befehl `\displaybreak` eingefügt werden, der einen Seitenumbruch an dieser Stelle hervorruft.

6.2 Weitere $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ -Befehle

6.2.1 Text

Mittels des Befehls

`\text{Normaler Text}`

kann temporär in den Textmodus geschaltet werden. Ähnlich funktioniert der Befehl

`\intertext{Normaler Text}`

nur daß vor und nach dem `\intertext` übergebenen Text Zeilenumbrüche eingefügt werden, dieser Text also auf eigenen Zeilen steht.

6.2.2 Matrizen

`Xmatrix`

Für Matrizen stehen dem Benutzer mehrere Umgebungen zur Verfügung, die sich gegenüber `array` durch besser gesetzten Leerraum unterscheiden und die selbst die begrenzenden Klammern setzen. Die neuen Umgebungen für Matrizen sind (zusammen mit ihren begrenzenden Klammern): `pmatrix ()`, `bmatrix []`, `Bmatrix { }`, `vmatrix | |` und `Vmatrix || ||`.

Beispiel:

`\begin{pmatrix}`

`a&b\\`

`c&d\\`

`\end{pmatrix}`

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

`matrix`

`smallmatrix`

Zusätzlich gibt es noch die Umgebungen `matrix`, die keine begrenzenden Klammern mitbringt, und `smallmatrix`, die eine kleinere Schrift verwendet und daher gut in Textzeilen verwendet werden kann.

6.2.3 Klammern

Manchesmal versagt $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ s automatischer Größenkontrollmechanismus für mit `\left` und `\right` erzeugte Klammerpaare, z. B.

$$\left[\sum_i a_i \left| \sum_j x_{ij} \right|^p \right]^{1/p}$$

oder

$$((a_1 b_1) - (a_2 b_2)) ((a_2 b_1) + (a_1 b_2))$$

Daher stellt $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ Befehle bereit, mit denen der Benutzer die Klammergröße bestimmt, sodaß das Ergebnis ansprechender aussieht:

$$\left[\sum_i a_i \left| \sum_j x_{ij} \right|^p \right]^{1/p}$$

oder

$$((a_1b_1) - (a_2b_2))((a_2b_1) + (a_1b_2))$$

Der Code für dieses Beispiel ist:

```
\begin{gather*}
\biggl[\sum_{i}a_i\bigl|\sum_{j}x_{ij}\bigr|^p\biggr]^{1/p}\backslash
\intertext{oder}
\bigl((a_1b_1)-(a_2b_2)\bigr)\bigl((a_2b_1)+(a_1b_2)\bigr)
\end{gather*}
```

Die $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ -Befehle sind `\big`, `\Big`, `\bigg` und `\Bigg`, gefolgt von 1 oder r für linke oder rechte Klammer sowie dem Klammerzeichen.

6.2.4 Akzente

Mit $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ sollte der Verwendung der Akzent-Befehle

```
\Hat \Check \Tilde \Acute \Grave \Dot \Ddot
\Breve \Bar \Vec
```

vorzug gegeben werden, da mit ihnen Mehrfachakzente korrekt positioniert werden, vgl. $\hat{\hat{A}}$ vs. $\hat{\hat{A}}$.

6.2.5 Weiteres

Der Befehl

```
\boxed{Formel}
```

bewirkt eine eingeschachtelte Formel.

```
\boxed{i^2=-1}  $i^2 = -1$ 
```

Entsprechend dem $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ -Befehl `\frac` stellt $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ den Befehl

```
\binom{Zähler}{Nenner}
```

bereit, mit dem Binomialkoeffizienten gesetzt werden können, z. B.

```
\begin{equation*}
\binom{n}{k} = \frac{n!}{k!(n-k)!}
\end{equation*}
```

Querverweise auf Gleichungen lassen sich bequem mit dem Befehl `\eqref` statt `\qref`

`\ref` realisieren, da bei `\eqref` automatisch die Klammern mitgesetzt werden.

$\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ bietet noch viele weitere Befehle, der interessierte Leser sei daher auf die $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ -Dokumentation [31] verwiesen.

Kapitel 7

Literaturverzeichnisse

In vielen Dokumenten wird Bezug auf andere Schriftstücke genommen. Dies geschieht üblicherweise über Literaturverzeichnisse, in denen ein Schlüssel und die exakte Referenz gegeben sind, so daß im Text nur der Schlüssel – und nicht immer die gesamte Referenz – angegeben werden muß.

Solange nur wenige Dokumente mit Literaturverzeichnissen unterhalten werden müssen, kann man lokale Literaturverzeichnisse verwenden (s. Abschn. 7.2), bei einer größeren Anzahl lohnt sich eigentlich fast immer die Erstellung einer Literaturdatenbank, die dann von jedem $\text{T}_{\text{E}}\text{X}$ -Dokument verwendet werden kann (s. Abschn. 7.3).

Weitere Ausführungen zu Literaturdatenbanken sind z. B. in [3] oder [28] zu finden.

7.1 Literaturbezüge im Text

Unabhängig davon, ob auf ein lokales Literaturverzeichnis in der $\text{T}_{\text{E}}\text{X}$ -Datei oder auf eine $\text{BIB}_{\text{T}_{\text{E}}\text{X}}$ -Literaturdatenbank zugegriffen wird, erfolgt der Bezug immer mit dem Befehl

```
\cite{Schlüssel}
```

Existiert also im Literaturverzeichnis ein Eintrag mit dem Schlüssel `kopka1`, so wird durch

```
\cite{kopka1}
```

Bezug auf ihn genommen.

Ebenso wie bei Querverweisen innerhalb des Dokuments sind *zwei* $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Läufe erforderlich, bis die Referenz korrekt aufgelöst wird.

7.2 Lokale Literaturverzeichnisse

Lokale Literaturverzeichnisse¹ werden mittels der `thebibliography`-Umgebung definiert, und zwar genau an der Stelle, an der im späteren Text das Literaturverzeichnis erscheinen soll.

¹Lokal bedeutet hier, daß für jedes Dokument manuell ein Literaturverzeichnis im Dokument angelegt wird, siehe dazu auch Abschnitt 7.3.

```

\begin{thebibliography}{Mustermarke}
\bibitem[Marke1]{Schlüssel1}Eintrag1
\bibitem[Marke2]{Schlüssel2}Eintrag2
...
\end{thebibliography}

```

Für die Schlüsselangaben im Literaturverzeichnis wird Platz in der Breite des Textes *Mustermarke* reserviert. Idealerweise sollte also der längste Schlüssel für den Parameter *Mustermarke* angegeben werden.

Die einzelnen Einträge in das Literaturverzeichnis erfolgen mit dem Befehl

```
\bibitem[Marke]{Schlüssel}Eintrag
```

Der Parameter *Schlüssel* gibt den Schlüssel an, über den per `\cite` bezug genommen werden kann. Der Text *Eintrag* enthält den Text, der im Literaturverzeichnis erscheinen soll. Standardmäßig ist die Marke eine in eckige Klammern eingeschlossene ganze Zahl, [1], [2], [3], ... Dieses Verhalten kann mit dem optionalen Parameter *Marke* geändert werden, dessen Text stattdessen zwischen den Klammern steht.

```

\begin{thebibliography}
\bibitem[1a]{kopka1}
  Kopka, Helmut: \emph{\LaTeX: Einführung,}
  2.~überarb. Aufl.,
  Bonn; Reading, Mass. [u.\,a.]: Addison-Wesley, 1996,
  ISBN 3--8273--1025--3
\bibitem[1b]{kopka2}
  Kopka, Helmut: \emph{\LaTeX: Ergänzungen -- mit einer
  Einführung in {\small METAFONT},}
  Bonn; Reading, Mass. [u.\,a.]: Addison-Wesley, 1995,
  ISBN 3--89319--665--X
\bibitem[1c]{kopka3}
  Kopka, Helmut: \emph{\LaTeX: Erweiterungen,}
  Bonn; Reading, Mass. [u.\,a.]:
  Addison-Wesley-Longman, 1997,
  ISBN 3--89319--666--8
\end{thebibliography}

```

7.3 Literaturdatenbanken mit BIB_TE_X

Treten häufig dieselben Literaturangaben auf, so wird es lästig, immer denselben Text in jede T_EX-Datei einzugeben. Da lohnt sich dann die Einrichtung einer Literaturdatenbank, auf deren Einträge ebenso mittels `\cite` bezug genommen werden kann, wie auf die Einträge eines lokalen Literaturverzeichnisses.

7.3.1 Das Datenbankformat

Die Datenbank-Datei wird – ebenso wie der L^AT_EX-Quelltext – mit einem ASCII-Editor bearbeitet, die Endung ist `.bib` – z. B. könnte `latex.bib` eine Literaturdatenbank zum Thema L^AT_EX sein. Die Einträge in der Datei haben alle die Form

```
@Literaturtyp{Referenz,
  Schlüssel_1 = {Wert_1},
  ...
  Schlüssel_n = {Wert_n}
}
```

Dabei ist *Literaturtyp* einer der in Tab. 7.1 angegebenen Eingabetypen und *Referenz* der Schlüssel, der beim `\cite`-Befehl angegeben wird. Für die An-

Typ	Beschreibung
article	für einen Artikel aus einem Journal oder einer Zeitschrift
book	für ein Buch aus einem Verlag
booklet	für ein Buch ohne Verlagsangabe
conference	für einen Artikel aus einem Tagungs- oder Konferenzbericht
inbook	für einen Buchauszug, etwa ein Kapitel oder bestimmte Seiten
incollection	für einen Buchauszug mit eigenem Titel
inproceedings	identisch mit <code>conference</code> , s. d.
manual	für eine technische Dokumentation
mastersthesis	für eine Diplomarbeit
misc	für Literaturstellen, die unter keinen der anderen Eingabetypen fallen
phdthesis	für eine Doktorarbeit
proceedings	für einen Tagungs- oder Konferenzbericht
techreport	für einen Bericht einer Hochschule, eines Forschungsinstituts u. ä., evtl. in Form einer herausgegebenen Serie mit einer laufenden Seriennummer
unpublished	für eine unveröffentlichte Arbeit

Tabelle 7.1: Literaturtypen von BIBTEX

gaben

```
Schlüssel = {Wert},
```

werden die in Tabelle 7.2 aufgeführten Schlüssel von BIBTEX berücksichtigt. Schlüssel, die nicht in dieser Tabelle auftauchen, werden von BIBTEX ignoriert, mögen aber für andere Zwecke sinnvoll sein².

Als Beispiel sollen dieselben Literaturangaben dienen, wie oben beim lokalen Literaturverzeichnis.

```
@book{kopka1,
  author = {Kopka, Helmut},
  title = {\LaTeX: Einf{"u}hrung},
  publisher = {Addison-Wesley},
```

²Z. B. kann der Schlüssel `url` von `bibtex2html` [32] ausgewertet werden. `bibtex2html` erzeugt aus bib-Dateien html-Dateien und fügt anwählbare Links auf die in `url` referenzierten Seiten ein.

Feld	Bedeutung
<code>address</code>	Verlagsanschrift, bei bekannteren Verlagen reicht die Angabe des Verlagsortes
<code>annotate</code>	Anmerkung, wird von <code>BIBTEX</code> ignoriert
<code>author</code>	Autorenname(n). „et al.“ wird durch <code>and others</code> erzeugt.
<code>booktitle</code>	Buchtitel, im Zusammenhang mit <code>@inbook</code> zu verwenden
<code>chapter</code>	Kapitelnummer, evtl. mit Kapitelüberschrift
<code>edition</code>	Auflagenummer
<code>editor</code>	Name des/Namen der Herausgeber
<code>howpublished</code>	Für Veröffentlichungen außerhalb eines Verlages
<code>journal</code>	Name eines Journals oder einer Zeitschrift
<code>key</code>	bestimmt die alphabetische Einordnung, falls Autor und Herausgeber fehlen
<code>month</code>	Monat der Veröffentlichungen
<code>note</code>	Zusatzinformationen, wird von <code>BIBTEX</code> ignoriert
<code>number</code>	Laufende Nummer eines Journals
<code>organization</code>	Organisation, die eine Tagung oder Konferenz ausgerichtet hat
<code>pages</code>	Seitenangabe
<code>publisher</code>	Verlagsname
<code>school</code>	Name einer Hochschule oder Universität
<code>series</code>	Name einer Buchserie
<code>title</code>	Artikel- oder Buchtitel
<code>type</code>	Typ eines Berichtes, „Forschungsbericht“ o. ä.
<code>volume</code>	Bandnummer
<code>year</code>	Jahr der Veröffentlichung

Tabelle 7.2: `BIBTEX`-Felder

```

year = {1996},
address = {Bonn; Reading, Mass. [u.\,a.]},
edition = {2. {\u}berarb.},
isbn = {3--8273--1025--3}
}
@book{kopka2,
  author = {Kopka, Helmut},
  title = {\LaTeX: Erg{\a}nzungen},
  publisher = {Addison-Wesley},
  year = {1995},
  address = {Bonn; Reading, Mass. [u.\,a.]},
  note = {Mit einer Einf{\u}hrung in {\small METAFONT}},
  isbn = {3--89319--665--X}
}
@book{kopka3,
  author = {Kopka, Helmut},

```


Stil	Beschreibung
plain	Eintragungen nach Autorennamen alphabetisch sortiert, laufende Nummern in eckigen Klammern als Kennzeichnung
unsrt	Eintragungen in der Reihenfolge der <code>\cite</code> - und <code>\nocite</code> -Befehle, Kennzeichnung wie bei <code>plain</code>
alpha	Reihenfolge wie bei <code>plain</code> , Kennzeichnung durch Abkürzung des Autorennamens + Jahreszahl
abbrv	Wie <code>plain</code> , nur wird soviel abgekürzt wie möglich

Tabelle 7.3: Die wichtigsten BIBTEX-Stile

```

title = {\LaTeX: Erweiterungen},
publisher = {Addison-Wesley-Longman},
year = {1997},
address = {Bonn; Reading, Mass. [u.\,a.]},
isbn = {3--89319--666--8}
}

```

7.3.2 L^AT_EX-Befehle für Literaturdatenbanken

An die Stelle der `thebibliography`-Umgebung bei lokalen Literaturverzeichnissen tritt bei der Verwendung von BIBTEX der Befehl

```
\bibliography{LitDB_1,LitDB2,...}
```

wobei die Endung `.bib` der Datenbank-Dateien *wegfallen muß* und zwischen den Einträgen *keine Leerzeichen* auftreten dürfen.

In das Literaturverzeichnis werden alle Literaturangaben aufgenommen, auf die innerhalb des L^AT_EX-Dokuments mit `\cite`-Befehlen Referenz genommen wird. Sollen auch bestimmte Angaben in das Literaturverzeichnis aufgenommen werden, auf die im laufenden Text nicht verwiesen wird, so kann dies mit dem Befehl

```
\nocite{Schlüssel}
```

erreicht werden. Dieser Befehl funktioniert genauso wie der Befehl `\cite`, nur wird kein Verweis im laufenden Text eingefügt, sondern nur der Eintrag ins Literaturverzeichnis gefordert.

Der Stil des Literaturverzeichnisses kann mit dem Befehl

```
\bibliographystyle{Stil}
```

beeinflusst werden. Die wichtigsten Stile sind in Tab. 7.3 angeführt. Stile nach eigenen Anforderungen lassen sich durch Aufruf von

```
tex makebst
```

erstellen, dabei wird alles interaktiv vom Benutzer abgefragt. Am Ende wird die fertige Stildatei (mit Endung `.bst`) bereitgestellt. Als Alternative hierzu existiert das Programm `custombib` [33].

7.3.3 BIB_TE_X-Durchlauf

Um alle Abhängigkeiten korrekt aufzulösen, sind mehrere Aufrufe von L^AT_EX und BIB_TE_X notwendig:

1. Der erste L^AT_EX-Durchlauf sucht alle benötigten Verweise aus der T_EX-Datei heraus und schreibt sie (neben anderen Informationen) in die `.aux`-Datei.
2. BIB_TE_X sucht die benötigten Verweise aus der `.aux`-Datei heraus und extrahiert die entsprechenden Literaturangaben aus der Datenbank (`.bib`). Das fertige Literaturverzeichnis wird in eine `.bbl`-Datei geschrieben, Fehler- und Statusmeldungen in eine `.blg`-Datei. Das Format des erzeugten Literaturverzeichnisses wird durch die ausgewählte Stildatei (`.bst`) bestimmt.
3. Der nun folgende L^AT_EX-Aufruf liest die `.bbl`-Datei ein und kann nun auch die korrekten Verweise in die `.aux`-Datei eintragen.
4. Der letzte L^AT_EX-Aufruf kann jetzt die korrekten Verweise in das Dokument einfügen.

Kurz gefaßt erreichen die folgenden vier Befehle eine korrekte Auflösung aller Referenzen:

```
latex Datei [.tex]
bibtex Datei
latex Datei [.tex]
latex Datei [.tex]
```

7.3.4 Mehrere Literaturverzeichnisse

L^AT_EX kann auch mehrere Literaturverzeichnisse erstellen und verwalten, inkl. separater Numerierung. Hierzu dient das Paket `multibib`.

Beispielsweise sollen zwei Literaturverzeichnisse angelegt werden, eine mit eigenen Arbeiten, eine zweite mit den Arbeiten von anderen Personen. Die eigenen Arbeiten sollen durch ein vorgesetztes E in den Zitierungen gekennzeichnet werden. Desweiteren seien die eigenen Arbeiten in `meinname.bib` und die restlichen Zitate in `andere.bib`.

Unglücklicherweise unterstützt dies `multibib` nicht direkt, daher muß ein kleiner Trick verwendet werden. Im Dokumentenkopf ist dazu

```
\usepackage[resetlabels]{multibib}
\makeatletter
\newcommand\biblabelprefix{}
\renewcommand\@biblabel[1]{[\biblabelprefix#1]}
\def\@bibitem#1{\item\if@filesw
\immediate\write\@auxout{\string\bibcite{#1}
{\biblabelprefix\the\value{\@listctr}}}\fi\ignorespaces}
\makeatother
\newcites{meine}{Mein Literaturverzeichnis}
\newcites{rest}{Allgemeines Literaturverzeichnis}
```

einzufügen und im eigentlichen Dokument – an der Stelle, wo das eigene Literaturverzeichnis erscheinen soll – wird dann

```
\renewcommand\biblabeledprefix{E}
\bibliographystylemy{unsrt_new}
\bibliographyselbst{meinname}
```

eingefügt und analog an der Stelle des zweiten Literaturverzeichnisses:

```
\renewcommand\biblabeledprefix{}
\bibliographystylemy{rest{unsrt_new}}
\bibliographyselbst{andere}
```

Nun kann wie gewohnt zitiert werden, lediglich mit dem Unterschied, das jetzt zwei verschiedene Zitierbefehle existieren, die sich auf die zwei Literaturverzeichnisse beziehen:

```
In meiner Arbeit\citemeine{Myname:1998} zeigte ich, daß das
Problem gelöst werden kann, im Gegensatz zu den
Arbeiten\citerest{Author1:1990,Author2:1995} die behaupten\dots
```

Die normalen Zitierbefehle werden lediglich um die oben definierten Namen *meine* und *rest* ergänzt.

Natürlich sind hiermit auch mehr als zwei Literaturverzeichnisse möglich.

7.3.5 bib-Dateien veröffentlichen

Die Pflege einer Datenbank mit eigenen Veröffentlichungen mittels BIB_TE_X hat noch einen weiteren Vorteil: Die .bib-Datei kann im Internet veröffentlicht werden und somit anderen das korrekte Zitieren erleichtern (sowie die eigenen Leistungen entsprechend darstellen). Insbesondere wenn die .bib-Datei bereits existiert, ist dies mit wenig Aufwand möglich.

Wenn im Internet zudem eine Webseite über die eigene Arbeit existiert oder diese komplett im Netz steht, dann kann zudem automatisch ein Hyperlink dorthin generiert werden, so daß der interessierte Leser mit wenig Aufwand an die Beschreibung oder das Werk selber kommen kann.

Hier soll das Vorgehen mit `bibtex2html` [32] vorgestellt werden. Als Beispiel dient hier ein Preprint, wie er z. B. in der Physik üblich ist. Nach dem Einreichen des Preprints hat jedes Dokument eine eindeutige URL, unter der es angesprochen werden kann, in diesem Beispiel `http://xxx.uni-augsburg.de/abs/quant-ph/0011107`. Diese URI wird nun als zusätzliche Information in der .bib-Datei angegeben, d. h. der Eintrag dazu erhält die Zeile

```
url = {http://xxx.uni-augsburg.de/abs/quant-ph/0011107},
```

Nun enthält die von `bibtex2html` generierte Datei bei diesem Eintrag einen Link auf den Preprint.

Kapitel 8

Indexerstellung

L^AT_EX kann zwar Indexregister nicht automatisch erstellen – wie z. B. Inhaltsverzeichnisse –, aber es unterstützt den Anwender bei dessen Erstellung. In diesem Kapitel soll gezeigt werden, wie mit Hilfe von L^AT_EX und dem Programm `makeindex` ein Index erzeugt werden kann.

Um Indexerstellung anzuschalten, muß im Vorspann der Befehl

```
\makeindex
```

stehen. Es empfiehlt sich daher, die Indexeinträge von vornherein gleich in den Quelltext einzubauen und später dann mit `\makeindex` die Indexerstellung zu aktivieren. Desweiteren wird für einige Befehle das Paket `makeidx` benötigt.

```
\usepackage{makeidx}
```

Bei jedem L^AT_EX-Durchlauf werden dann die Indexeinträge in der Reihenfolge ihres Auftretens zusammen mit der Seitennummer in eine Datei mit der Endung `.idx` geschrieben. Diese Datei dient dem Programm `makeindex` als Grundlage. Zur weiteren Vertiefung wird [3, 28] empfohlen.

8.1 Indexeinträge

Der L^AT_EX-Befehl für einen Indexeintrag ist

```
\index{Eintrag}
```

Im einfachsten Fall ist *Eintrag* Klartext, der im Indexregister erscheinen soll. Soll ein Untereintrag angegeben werden, so werden Haupt- und Untereintrag (und evtl. Unteruntereintrag) durch ! voneinander getrennt:

```
\index{Haupteintrag!Untereintrag}
```

```
\index{Haupteintrag!Untereintrag!Unteruntereintrag}
```

Für L^AT_EX-Befehle bietet sich an, sie alphabetisch einzuordnen, allerdings ohne den führenden Backslash. Für solche Fälle können im *Eintrag* zwei durch @ getrennte Felder eingegeben werden. Der Eintrag wird dann so einsortiert wie im ersten Feld und ausgegeben wie im zweiten Feld, z. B.

```
\index{cite@\verb+\cite+}
\index{LaTeX@\verb+\LaTeX+}
```

Eine weitere Möglichkeit des Eintrags ist, einen Befehl aufzurufen. Dazu besteht *Eintrag* wiederum aus zwei Feldern, diesmal durch | getrennt. Der Eintrag wird so einsortiert wie im ersten Feld. Im zweiten Feld wird \LaTeX -Befehl ohne den führenden Backslash eingegeben, der dann mit der Seitenzahl als Parameter aufgerufen wird. An einem Beispiel wird dies deutlicher:

```
\index{Indexregister|textit}
```

In diesem Beispiel wird die zugehörige Seitenangabe kursiv gesetzt. Ein Spezialfall ist der Befehl `\see`. Wird der Indexeintrag mittels

```
\index{Indexerstellung|see{Indexregister}}
```

getätigt, so erscheint im Indexregister „Indexerstellung, *siehe* Indexregisters“. Soll eines der Sonderzeichen !, @ und | als Textzeichen auftreten, so ist es mit " zu maskieren: "!, "@, "|. Dies gilt ebenfalls für " selbst: "". Sollen die mit " eingeleiteten Umlaute verwendet werden, so kann mittels einer *Formatänderungsdatei* (s. u.) ein anderes Zeichen zur Maskierung bestimmt werden.

8.2 Indexerzeugung und -ausgabe

Das Programm `makeindex` wird mit der von \LaTeX erzeugten `.idx` aufgerufen,

```
makeindex Datei [.idx]
```

liest die unsortierten Indexeinträge ein, sortiert sie und schreibt den fertig formatierten Index in eine Datei mit der Endung `.ind`. Die wichtigsten Optionen für `makeindex` sind in Tab. 8.1 wiedergegeben. Der auf diese Weise erzeugte Index wird an der Stelle des \LaTeX -Befehls

```
\printindex
```

in das Dokument eingefügt.

Option	Bedeutung
-l	Leerzeichen bleiben beim Sortiervorgang unberücksichtigt (L etter ordering)
-c	Überflüssige Leerzeichen bleiben beim Sortiervorgang unberücksichtigt (C ompress blanks)
-g	Deutsches Ordnungsschema für mit " eingegebene Umlaute. Voraussetzung: Verwendung einer Formatänderungsdatei, s. d. (G erman ordering)
-s	Gefolgt von dem Namen einer <i>Formatänderungsdatei</i> , (s. d.), die zusätzlich eingelesen wird (S tyle declaration)

Tabelle 8.1: `makeindex`-Optionen

Eine *Formatänderungsdatei* besteht aus einer Liste von Paaren der Form *Schlüssel Wert*. Die wichtigsten sind die Angaben zur Änderung der Sonderzeichen, die in Tab. 8.2 mit dem Beispiel der Standardattribute wiedergegeben sind. Für weitere Parameter verweise ich auf die Dokumentation von `makeindex`. Als Beispieldatei empfehle ich `gind.ist`.

Schlüssel-Wert-Paar	Beschreibung
<code>quote</code> <code>'"</code>	bestimmt das Maskierungszeichen
<code>level</code> <code>'!</code>	bestimmt das Trennzeichen für Untereinträge
<code>actual</code> <code>'@'</code>	bestimmt das lexikalische Zuordnungszeichen
<code>encap</code> <code>' '</code>	bestimmt das Pseudobefehlszeichen für die Seitenzahlformatierung

Tabelle 8.2: Die wichtigsten Schlüsselwörter für Formatänderungsdateien am Beispiel der Standardattribute

Kapitel 9

pdfT_EX

Das *Portable Document Format (PDF)* hat sich aufgrund der freien Verfügbarkeit des *Acrobat Readers* zu einem Format entwickelt, das auf nahezu jeder Plattform betrachtet werden kann. Mit pdfT_EX haben HÀN THÉ THÀNH und SEBASTIAN RAHTZ ein T_EX-System entwickelt, mit dem die Ausgabe im PDF-Format erzeugt werden kann, und das zudem noch kompatibel zu DONALD E. KNUTHs original T_EX-System ist.

9.1 Aufruf von pdfT_EX

Um pdfL^AT_EX statt L^AT_EX zu verwenden ist einfach der Befehl `latex` durch den Befehl `pdflatex` zu ersetzen, wie in

```
pdflatex beispiel.tex
```

Auch für plain T_EX gibt es Ersatz, nämlich statt `tex` den Befehl `pdftex`. Unter Umständen kann die Verwendung der Standard-T_EX-Schriften im Zusammenhang mit PDF-Ausgabe zu Beeinträchtigungen der Darstellung führen. Daher empfehle ich die Verwendung von PostScript-Schriften, was z. B. mit

```
\usepackage{times,mathptmx}
```

erreicht werden kann.

9.2 Grundlegende pdfT_EX-Befehle

Mit der Variable `\pdfoutput` wird festgelegt, ob die Ausgabe als PDF- oder DVI-Datei erfolgen soll. Dabei legen positive Werte PDF als Ausgabeformat fest, andere Werte DVI. Mit

```
\pdfoutput=1
```

erfolgt somit die Ausgabe als PDF-Datei.

Das PDF-Format unterstützt verschiedene Komprimierungsstufen. Mit der Variable `\pdfcompresslevel` wird die Komprimierungsstufe der Ausgabedatei angegeben. 0 bedeutet keine Kompression, 9 bedeutet maximale Kompression. Die heutigen Rechner sind bereits so leistungsfähig geworden, daß auch bei maximaler Kompression kaum erhöhter Zeitbedarf zu merken ist. Somit sollte ein

```
\pdfcompresslevel=9
```

in der \TeX -Datei stehen.

Die Ausgaberoutine von pdf \TeX wurde gegenüber dem Original dahingehend verändert, daß die einzelnen Buchstaben leicht gedehnt oder gestaucht werden dürfen, um annähernd gleiche Wortzwischenräume über das gesamte Dokument zu erhalten. Mit dem Befehl `\pdfadjustspacing` kann dieses Verhalten gesteuert werden. Ruft man ihn mit dem Argument 1 auf, so wird es nahezu unterbunden, und das Ergebnis entspricht ziemlich genau dem, was K_NU_TH_S \TeX auch ausgibt. Ruft man ihm dagegen mit dem Argument 2 auf,

```
\pdfadjustspacing2
```

dann macht pdf \TeX optimal Gebrauch von dieser neuen Möglichkeit.

9.3 Dokumenten-Information

In einem PDF-Dokument können Angaben über den Autor, das Erstellungsdatum, Schlüsselwörter, eine Kurzbeschreibung und einiges mehr gespeichert sein. Diese Informationen lassen sich mit dem Befehl `\pdfinfo` angeben, wie in

```
\pdfinfo{%
  /Author (Helge Kreuzmann,Carsten Luckmann)
  /Title (LBuch.pdf)
  /CreationDate (D:20000418213200)
  /Subject (Praktisches LaTeX)
  /Keywords (LaTeX)
}
```

9.4 Hyperlinks

PDF unterstützt – ebenso wie HTML, die Sprache des WWW – *Hyperlinks*, über die man leicht zu einer anderen Stelle desselben Dokuments oder auch zu einer Internet-Seite springen kann. Zunächst wird das Paket `hyperref` vorgestellt, das automatisch Hyperlinks für alle Verweise, Inhaltsverzeichnis-Einträge usw. erzeugt. Anschließend wird noch erläutert, wie manuell Hyperlinks erzeugt werden können; dies sollte jedoch wenn möglich vermieden werden.

9.4.1 Das hyperref-Paket

Das Paket `hyperref` automatisiert weitgehend die Erstellung von Hyperlinks in \LaTeX Dokumenten. Wie andere Pakete auch, wird es mit

```
\usepackage[pdftex=true]{hyperref}
```

eingebunden, die Option `pdftex` macht `hyperref` kompatibel zu `pdftex`. Da `hyperref` viele \LaTeX -Befehle undefiniert, sollte es als letztes Paket geladen werden. Die richtige Position kann am besten durch Ausprobieren erhalten werden.

Das `hyperref`-Paket kennt mehrere Optionen, die wichtigsten sind in Tab. 9.1 aufgeführt. Weitere Optionen, z. B. wie die Farbe verschiedener Links festgelegt werden kann, sind in der `hyperref`-Dokumentation [35] beschrieben.

Option	Bedeutung
<code>hyperindex=true</code>	Hyperlinks auch vom Index aus.
<code>bookmarks=true</code>	Inhaltsverzeichnis für den Acrobat Reader.
<code>bookmarksnumbered=true</code>	Inhaltsverzeichnis mit Abschnittsnummern.
<code>colorlinks=true</code>	Links farbig kennzeichnen.

Tabelle 9.1: hyperref-Optionen

9.4.2 Manuelle Links

Das Ziel eines solchen Sprungs wird mit dem Befehl `\pdfdest` definiert

```
\pdfdest < num n | name Referenz > appearance
```

Dabei wird dieser Zielpunkt durch eine Nummer *n* oder einen symbolischen Namen *Referenz* gekennzeichnet. Der Parameter *appearance* gibt den Vergrößerungsfaktor an und kann die in Tab. 9.2 angegebenen Werte annehmen.

Schlüsselwort	Bedeutung
<code>fit</code>	Seite paßt ins Fenster
<code>fith</code>	Seite paßt horizontal ins Fenster
<code>fitv</code>	Seite paßt vertikal ins Fenster
<code>fitb</code>	„Bounding Box“ paßt ins Fenster
<code>fitbh</code>	„Bounding Box“ paßt horizontal ins Fenster
<code>fitbv</code>	„Bounding Box“ paßt vertikal ins Fenster
<code>xyz</code>	Momentane Vergrößerung beibehalten

Tabelle 9.2: Argumente für den Parameter *appearance* des Befehls `\pdfdest`

Der Link selbst wird dann mit dem Befehl `\pdfannotlink` angelegt. Im einfachsten Falle lautet er einfach

```
\pdfannotlink Aktion
```

wobei *Aktion* die in Tab. 9.3 angegebenen Werte annehmen kann.

<i>Aktion</i>	Ziel
<code>page <i>n</i></code>	Seite <i>n</i>
<code>goto num <i>n</i></code>	Link Nr. <i>n</i>
<code>goto name <i>Referenz</i></code>	Link <i>Referenz</i>
<code>goto file <i>Datei</i></code>	Öffne Datei <i>Datei</i>
<code>usr { /URI (<i>URI</i>) }</code>	Öffne das Dokument mit dem URI <i>URI</i>

Tabelle 9.3: Aktionen für Hyperlinks

Im Anschluß an diesen Befehl kommt der Linktext, der durch

`\pdfendlink`

beendet wird.

9.4.3 Inhaltsverzeichnis (manuell)

Inhaltsverzeichnisse in T_EX-Manier beherrscht pdfT_EX natürlich auch, darüberhinaus aber auch hyperlink-artige Inhaltsverzeichnisse, bei denen man durch Klicken auf eine Überschrift direkt bei dem Kapitel landet, wie z. B. im Acrobat Reader. Mit `hyperref` geschieht dies automatisch, ohne muß man ein wenig Hand anlegen. Der Befehl dazu lautet

```
\pdfoutline Aktion count n {Text}
```

Aktion bezeichnet wieder eine der in Tab. 9.3 genannten Aktionen, *Text* sollte idealerweise die Überschrift des Kapitels, des Abschnitts usw. enthalten. *count* kennzeichnet die Anzahl direkter Untereinträge, also bei Kapiteln die Anzahl der Abschnitte in diesem Kapitel, bei Abschnitten die Anzahl der Unterabschnitte.

9.5 Einfach umschalten zwischen PDF- und DVI-Ausgabe

An manchen Stellen mag es passieren, daß T_EX-Code speziell für DVI oder speziell für PDF geschrieben werden muß. In dem Fall wäre es wünschenswert, wenn man im Dokument eine Abfrage machen könnte, und in Abhängigkeit des Ergebnisses die Befehle für DVI- oder PDF-Ausgabe ausführen könnte. Genug des Konjunktivs, unter

<http://www.itp.uni-hannover.de/~luckmann/download/>

steht das Paket `ifpdf` zum Download bereit. Dieses Paket bietet genau diese Möglichkeit und gewährleistet zusätzlich Kompatibilität zu Standard-L^AT_EX. Es wird mit

```
\usepackage[pdf]{ifpdf}
```

oder

```
\usepackage[dvi]{ifpdf}
```

eingebunden, je nachdem, ob PDF oder DVI-Ausgabe erzeugt werden soll. Desweiteren werden die zwei Umgebungen `pdfcode` und `dvice` bereitgestellt, die Code spezifisch für PDF bzw. DVI beinhalten¹. Wechseln zwischen PDF und DVI-Ausgabe geht nunmehr einfach durch Wechseln der Option für `ifpdf`.

¹Für ausgabeabhängige Definitionen im Dokumentenkopf wird dagegen das Konstrukt `\ifpdf` mit `\fi` benötigt, um z. B. nur bei der pdf-Ausgabe das Paket `hyperref` einzubinden.

Sollten weitere Tips & Tricks vorhanden sein, so wäre ich für einen Hinweis dankbar. Meine e-mail-Adresse steht vorne im Dokument. Ansonsten bleibt mir nur

Viel Erfolg

zu wünschen.

Literaturverzeichnis

- [1] Knappen et al. *LaTeX 2ε-Kurzbeschreibung Version 1.2*. ftp.dante.de.
- [2] Beschreibung zur LaTeX-Entwicklung inkl. LaTeX 3 unter www.latex-project.org.
- [3] Michael Goossens, Frank Mittelbach und Alexander Samarin. *Der LaTeX-Begleiter*. Addison-Wesley 1994. ISBN 3-89319-646-3.
- [4] E. Schlegl J. Knappen, H. Partl und I. Hyna. *LaTeX-Kurzbeschreibung. Version 3*; unter ftp.dante.de.
- [5] Dokumentation zum Paket german entweder lokal oder über ftp.dante.de.
- [6] Dokumentation zum graphicx-Paket unter `/usr/share/texmf/doc/latex/graphics` oder unter <ftp://ftp.dante.de/tex-archive/help/Catalogue/entries/graphicx.html>.
- [7] Goossens, Rahtz und Mittelbach. *The LaTeX graphics companion*. Addison-Wesley 1997.
- [8] Stephanie Hinrichs und Günther Beulshausen. *Gutenberg digital*. c't magazin für computer technik **15**, 274 (1997).
- [9] Download von dante über <ftp://ftp.dante.de/tex-archive/help/Catalogue/entries/foiltex.html>.
- [10] Homepage vom Acrobat Reader unter <http://www.adobe.com/products/acrobat/readstep.html>.
- [11] Homepage von ppower unter <http://www-sp.iti.informatik.tu-darmstadt.de/software/ppower4/>.
- [12] Homepage von texpower: <http://lrb.cs.uni-dortmund.de/%7Elehmke/TeXPower/>.
- [13] Kofler. *Linux*. Addison-Wesley 1996. ISBN 3-8273-1033-4; Hervorragendes Linux-Buch mit LaTeX, TclTk, Emacs(-Programmierung) uvm.
- [14] Homepage von xtem: http://www.iwd.uni-bremen.de/xtem.v8/xtem.texmenu_eng.html.
- [15] ts (TeXShell for X-Window) kann auf dante in <ftp://ftp.dante.de/tex-archive/systems/unix/texshell> gefunden werden.

- [16] Homepage von kTeXshell: <http://kTeXshell.sourceforge.net/>.
- [17] 1998. CD.
- [18] MiKTeX Verzeichnis auf Dante unter <ftp://ftp.dante.de/tex-archive/systems/win32/miktex/>.
- [19] Homepage von WinTeX 97: <http://home.t-online.de/home/mmuecke/>.
- [20] Dallheimer. c't magazin für computer technik.
- [21] Bauer et al. *S.u.S.E. Linux 4.2*. S.u.S.E. Gesellschaft für Systementwicklung 1996. S.u.S.E GmbH Gesellschaft für Software- Systementwicklung, <http://www.suse.de>.
- [22] RRZN. *UNIX - Eine Einführung*. RRZN achte überarbeitete edition 1994. Revision B.
- [23] Homepage von pstoedit:
<http://www.geocities.com/SiliconValley/Network/1958/pstoedit/>.
- [24] Homepage von xfig unter <http://www.xfig.org/>.
- [25] Die neuste Version von Image Magick gibts unter <http://www.wizards.dupont.com/cristy>.
- [26] Die lokale Dokumentation ist unter `/usr/share/texmf/doc` (SuSE) oder `/usr/share/texmf/source/` (Debian) installiert. Für Debian ist hierzu das Paket tetex-src nötig.
- [27] Homepage von texdoctk unter <http://www.geophysik.uni-frankfurt.de/%7Eruedas/progs.html>.
- [28] Helmut Kopka. *L^AT_EX: Einführung*. L^AT_EX. Addison-Wesley Bonn; Reading, Mass. [u. a.] 2. edition 1996.
- [29] Helmut Kopka. *L^AT_EX: Ergänzungen*. L^AT_EX. Addison-Wesley Bonn; Paris; Reading, Mass. [u. a.] 1995. Mit einer Einführung in METAFONT.
- [30] Helmut Kopka. *L^AT_EX: Erweiterungen*. L^AT_EX. Addison-Wesley Bonn; Reading, Mass. [u. a.] 1997.
- [31] American Mathematical Society. *AMS-L^AT_EX User's Guide* November 1996.
- [32] z. B. mit bibtex2html, das unter <http://www.lri.fr/%7Efilliatr/bibtex2html/> erhältlich ist. Damit sind auch Links direkt aus der bib-Datei auslesbar.
- [33] Eigene Stildateien können mittels des Pakets custom-bib erstellt werden: <http://www.uni-koblenz.de/%7Etexadmin/texmf/doc/html/latex/contrib/custom-bib/custom-bib.html>.
- [34] Hàn Thé Thàn, Sebastian Rahtz und Hans Hagen. *The pdfT_EX Manual* März 1999.
- [35] Sebastian Rahtz. *Hypertext marks in L^AT_EX: the hyperref package* Juni 1998.