

L^AT_EX 2_ε-Kurzbeschreibung

Jörg Knappen*
Hubert Partl†
Elisabeth Schlegl‡
Irene Hyna§

Version 1.1
20. Oktober 1994

L^AT_EX [1, 2] ist ein Textsatzsystem, das sich insbesondere für die „druckreife“ Erstellung von wissenschaftlichen Dokumentationen eignet, die mathematische Formeln enthalten. Es kann aber auch für viele andere Arten von Schriftstücken verwendet werden, von einfachen Briefen bis zu kompletten Büchern. L^AT_EX baut auf T_EX [6, 7] auf.

Die vorliegende Kurzbeschreibung basiert auf L^AT_EX Version L^AT_EX 2_ε vom 1. Juni 1994 und sollte für die viele Anwendungen von L^AT_EX ausreichend sein. Eine vollständige Beschreibung von L^AT_EX enthält das *L^AT_EX-Manual* [1].

L^AT_EX ist an vielen verschiedenen Groß- und Mikro-Rechnern installiert. Wie der Zugriff auf L^AT_EX in einer bestimmten Installation erfolgt und welche Komponenten dort verfügbar sind, ist im jeweiligen *Local Guide* [3] beschrieben.

*Zentrum für Datenverarbeitung, Johannes Gutenberg-Universität Mainz

†Zentraler Informatikdienst der Universität für Bodenkultur Wien

‡EDV-Zentrum der Karl-Franzens-Universität Graz

§Bundesministerium für Wissenschaft und Forschung Wien

Diese Beschreibung wurde mit L^AT_EX 2_ε in der Document Class für Artikel mit dem Paket für deutsche Sprache und mit den L^AT_EX-Befehlen für Titel, Inhaltsverzeichnis, Abbildungen, Tabellen, Literaturangaben u. dgl. erstellt. Sie dient also gleichzeitig als Beispiel für die von L^AT_EX standardmäßig unterstützten Layouts. Die L^AT_EX 2_ε-Eingabefiles sind am Server ftp.dante.de (Heidelberg) verfügbar.

Ich danke Michael Hofmann, Rainer Schöpf und Stefan Steffens für Tips, Anmerkungen und Korrekturen zur L^AT_EX 2_ε-Fassung dieser Kurzanleitung (J.K.).

Inhaltsverzeichnis

1	Allgemeines	6
1.1	The Name of the Game	6
1.1.1	TEX	6
1.1.2	LATEX	6
1.1.3	LATEX 2 ϵ	6
1.2	Grundkonzept	6
1.2.1	Autor, Designer und Setzer	6
1.2.2	Layout-Design	7
1.2.3	Vor- und Nachteile	7
1.3	Eingabefile	8
1.3.1	Leerstellen	8
1.3.2	Spezielle Zeichen	8
1.3.3	LATEX-Befehle	8
1.3.4	Kommentare	9
1.3.5	Aufbau	9
1.4	Layout	11
1.4.1	Document Class	11
1.4.2	Pakete	11
1.4.3	Page Style	12
2	Setzen von Text	14
2.1	Zeilen- und Seiten-Umbruch	14
2.1.1	Blocksatz	14
2.1.2	Silbentrennung	15
2.2	Spezielle Zeichen	16
2.2.1	Anführungszeichen	16
2.2.2	Binde- und Gedankenstriche	16
2.2.3	Punkte	16
2.2.4	Ligaturen	17
2.2.5	Akzente und spezielle Buchstaben	17
2.3	Abstände	17
2.4	Deutschsprachige Texte	19
2.4.1	Silbentrennung	19
2.4.2	Umlaute und scharfes s	19
2.4.3	Anführungszeichen	19
2.4.4	Sonstige Befehle	20
2.4.5	Überschriften und Datumsangaben	20
2.5	Kapitel und Überschriften	20
2.6	Fußnoten	22
2.7	Hervorgehobene Wörter	22
2.8	Umgebungen	23
2.8.1	Zitate (quote, quotation, verse)	23
2.8.2	Listen (itemize, enumerate, description)	23
2.8.3	Flattersatz (flushleft, flushright, center)	23
2.8.4	Direkte Ausgabe (verbatim, verb)	26

2.8.5	Abbildungen (figure)	26
2.8.6	Tabellen (table)	27
2.8.7	Tabulatoren (tabbing)	28
2.8.8	Eigentliche Tabellen (tabular)	28
3	Setzen von mathematischen Formeln	29
3.1	Allgemeines	29
3.2	Elemente in mathematischen Formeln	30
3.3	Nebeneinander Setzen	33
3.4	Übereinander Setzen	33
3.5	Liste der mathematischen Symbole	35
4	Spezialitäten	40
4.1	Schriftarten und -größen (Fonts)	40
4.2	Abstände	41
4.2.1	Zeilenabstand	41
4.2.2	Spezielle horizontale Abstände	41
4.2.3	Spezielle vertikale Abstände	42
4.3	Briefe (letter)	43
4.4	Literaturangaben	43
4.5	Robuste und zerbrechliche Befehle	44
4.6	Kompatibilität zu L ^A T _E X2.09	44

Abbildungsverzeichnis

1	Ein minimales L ^A T _E X 2 _ε -File	10
2	Aufbau eines Artikels	10
3	Beispiel für <code>itemize</code>	24
4	Beispiel für <code>enumerate</code>	24
5	Beispiel für <code>description</code>	25
6	Linksbündig, rechtsbündig und zentriert	25
7	R. Black, Ganz in weiß	27
8	Aufbau einer <code>table</code> -Umgebung	27
9	Brief von H. P. an E. S.	43

Tabellenverzeichnis

1	Document Classes	11
2	Document Class Options	12
3	Packages (eine Auswahl)	13
4	Akzente und spezielle Buchstaben	18
5	Überschriften	21
6	Mathematische Akzente	35
7	Große griechische Buchstaben	35
8	Kleine griechische Buchstaben	36
9	Verschiedene sonstige Symbole	36
10	„Große“ Operatoren	36
11	Binäre Operatoren	37
12	Relationen	37
13	Negationen	38
14	Pfeile	38
15	Linke Klammern	38
16	Rechte Klammern	38
17	Synonyme	39
18	Nicht-mathematische Symbole	39
19	Schriftarten	40
20	Schriftgrößen	41
21	Einheiten für Längenangaben	42
22	Horizontale Abstände	42
23	Vertikale Abstände	42

1 Allgemeines

1.1 The Name of the Game

1.1.1 T_EX

T_EX (sprich „Tech“, kann auch „TeX“ geschrieben werden) ist ein Computer-Programm von Donald E. Knuth [6, 7]. Es dient zum Setzen und Drucken von Texten und mathematischen Formeln.

1.1.2 L^AT_EX

L^AT_EX (sprich „Lah-tech“ oder „Lej-tech“, kann auch „LaTeX“ geschrieben werden) ist ein sogenanntes Makro-Paket von Leslie Lamport [1, 2], das T_EX verwendet. Es ermöglicht dem Autor eines Textes, sein Schriftstück in einfacher Weise unter Verwendung eines der vorgefertigten Layouts in Buchdruck-Qualität zu setzen und auszudrucken.

1.1.3 L^AT_EX 2_ε

L^AT_EX 2_ε (sprich „L^AT_EX zwei e“) ist die aktuelle Version von L^AT_EX seit dem 1. Juni 1994. Wenn hier von L^AT_EX gesprochen wird, so trifft das gesagte sowohl auf die alte Version L^AT_EX 2.09 als auch auf L^AT_EX 2_ε zu, wenn von L^AT_EX 2_ε die Rede ist, gilt das gesagte nur für L^AT_EX 2_ε.

1.2 Grundkonzept

1.2.1 Autor, Designer und Setzer

Für eine Publikation übergibt der Autor dem Verleger üblicherweise ein maschinengeschriebenes Manuskript. Der Buch-Designer des Verlages entscheidet dann über das Layout des Schriftstücks (Länge einer Zeile, Schriftart, Abstände vor und nach Kapiteln usw.) und schreibt dem Setzer die dafür notwendigen Steuerdaten dazu.

L^AT_EX ist sozusagen der Buch-Designer, T_EX ist sein Setzer. Die eingegebenen L^AT_EX-Befehle werden in um Stufen niedrigere T_EX-Setzbefehle übersetzt.

Ein menschlicher Buch-Designer erkennt die Absichten des Autors (z. B. Kapitel-Überschriften, Zitate, Beispiele, Formeln . . .) meistens auf Grund seines Fachwissens aus dem Inhalt des Manuskripts. L^AT_EX dagegen ist „nur“ ein Programm und benötigt daher zusätzliche Informationen vom Autor, die die logische Struktur des Textes angeben. Diese Informationen werden in Form von sogenannten „Befehlen“ innerhalb des Textes angegeben.

Im Gegensatz dazu steht ein optischer Entwurf eines Schriftstückes mit Textverarbeitungsprogrammen wie z. B. WordStar. In diesem Fall legt der Autor das Layout des Textes bei der interaktiven Eingabe fest. Dabei sieht er am Bildschirm das, was auch auf der gedruckten Seite stehen wird. Solche Systeme, die optische Entwürfe unterstützen, werden auch WYSIWYG-Systeme („what you see is what you get“) genannt.

Bei \LaTeX sieht der Autor beim Schreiben des Eingabefiles in der Regel noch nicht, wie der Text nach dem Formatieren aussehen wird. Er kann aber durch Aufruf des entsprechenden Programms jederzeit einen Probe-Ausdruck seines Schriftstücks – auf dem Bildschirm oder auf dem Drucker – machen und danach sein Eingabefile entsprechend korrigieren und die Arbeit fortsetzen.

1.2.2 Layout-Design

Typographisches Design ist ein Handwerk, das erlernt werden muß. Ungeübte Autoren machen oft gravierende Formatierungsfehler. Fälschlicherweise glauben viele Laien, daß Buchdruck-Design vor allem eine Frage der Ästhetik ist – wenn das Schriftstück vom künstlerischen Standpunkt aus schön aussieht, dann ist es schon gut „designed“. Da Schriftstücke jedoch gelesen und nicht in einem Museum aufgehängt werden, sind die leichtere Lesbarkeit und bessere Verständlichkeit wichtiger als das schöne Aussehen.

Beispiele: Die Schriftgröße und Numerierung von Überschriften soll so gewählt werden, daß die Struktur der Kapitel und Unterkapitel klar erkennbar ist. Die Zeilenlänge soll so gewählt werden, daß anstrengende Augenbewegungen des Lesers vermieden werden, nicht so, daß sie das Papier möglichst schön ausfüllt.

Mit interaktiven optischen Entwurfsystemen erzeugen Autoren im allgemeinen ästhetisch schöne, aber schlecht strukturierte Schriftstücke. \LaTeX verhindert solche Formatierungsfehler, indem es den Autor dazu zwingt, die logische Struktur des Textes anzugeben, und dann automatisch das dafür am besten geeignete Layout verwendet.

1.2.3 Vor- und Nachteile

\LaTeX zeichnet sich gegenüber anderen Textverarbeitungsprogrammen vor allem durch die folgenden Vorteile aus:

- Es stehen mehrere professionell gestaltete Layouts zur Verfügung, mit denen die Schriftstücke tatsächlich „wie gedruckt“ aussehen.
- Das Setzen von mathematischen Formeln ist besonders gut unterstützt.
- Der Anwender muß nur wenige, leicht verständliche Befehle angeben, die die logische Struktur des Schriftstücks betreffen, und braucht sich um die Details der drucktechnischen Gestaltung (fast) nicht kümmern.
- Auch komplexe Strukturen wie Fußnoten, Literaturangaben, Inhaltsverzeichnisse, Tabellen u. v. a. und sogar einfachere Zeichnungen können ohne großen Aufwand erstellt werden.

\LaTeX hat freilich auch Nachteile:

- Der Betriebsmittelverbrauch (Rechenzeit und Speicherplatz) am Computer ist höher als bei primitiveren Textverarbeitungsprogrammen.
- Innerhalb der von \LaTeX unterstützten Dokument-Layouts können zwar einzelne Parameter leicht variiert werden, grundlegende Abweichungen

Heute ist der 35. Mai 1987. Oder:
 Heute ist der 35. Mai 1987. Falsch
 ist: Am 35. Mai 1987 regnet es.
 Richtig: Am 35. Mai 1987 scheint
 die Sonne. Oder: Am 35. Mai 1987
 schneit es.

Heute ist der `\today`.
 Oder: Heute ist der `\today` .
 Falsch ist:
 Am `\today` regnet es.
 Richtig:
 Am `\today{}` scheint die Sonne.
 Oder: Am `\today\` schneit es.

Manche Befehle haben Parameter, die zwischen geschwungenen Klammern angegeben werden müssen. Manche Befehle haben Parameter, die weggelassen oder zwischen eckigen Klammern angegeben werden können. Manche Befehle haben Varianten, die durch das Hinzufügen eines Sterns an den Befehlsnamen unterschieden werden.

Geschwungene Klammern können auch dazu verwendet werden, Gruppen (groups) zu bilden. Die Wirkung von Befehlen, die innerhalb von Gruppen oder Umgebungen (environments) angegeben werden, endet immer mit dem Ende der Gruppe bzw. der Umgebung. Im obigen Beispiel ist `{}` eine leere Gruppe, die außer der Beendigung des Befehlsnamens `today` keine Wirkung hat.

1.3.4 Kommentare

Alles, was hinter einem Prozentzeichen (%) steht (bis zum Ende der Eingabezeile), wird von \LaTeX ignoriert. Dies kann für Notizen des Autors verwendet werden, die nicht oder noch nicht ausgedruckt werden sollen.

Das ist ein Beispiel.

Das ist ein % dummes
 % Besser: ein lehrreiches <----
 Beispiel.

1.3.5 Aufbau

Der erste Befehl in einem \LaTeX 2_{ϵ} -Eingabefile muß der Befehl

```
\documentclass
```

sein (siehe Abschnitt 1.4.1). Danach können weitere Definitionen folgen, die für das gesamte Schriftstück gelten sollen. Mit dem Befehl

```
\begin{document}
```

beginnt das Setzen des Schriftstücks. Nun folgen der Text und alle \LaTeX -Befehle, die das Ausdrucken des Schriftstücks bewirken. Die Eingabe muß mit dem Befehl

```
\end{document}
```

beendet werden. Falls nach diesem Befehl noch Eingaben folgen, werden sie von \LaTeX ignoriert.

Abbildung 1 zeigt ein *minimales* \LaTeX 2_{ϵ} -File. Ein etwas komplizierteres File ist in Abbildung 2 skizziert.

```
\documentclass{article}
\begin{document}
Small is beautiful.
\end{document}
```

Abbildung 1: Ein minimales L^AT_EX 2_ε-File

```
\NeedsTeXFormat{LaTeX2e}
\documentclass[11pt,a4paper]{article}
\usepackage{german,a4}
\author{H.~Partl}
\title{"Uber kurz oder lang}

\begin{document}
\maketitle
\begin{abstract}
Beispiel f"ur einen wissenschaftlichen Artikel
in deutscher Sprache.
\end{abstract}
\tableofcontents

\section{Start}

Hier beginnt mein sch"ones Werk\dots

\section{Ende}

\dots\ und hier endet es.

\end{document}
```

Abbildung 2: Aufbau eines Artikels

1.4 Layout

1.4.1 Document Class

Zu Beginn des Eingabefiles muß das Layout mit

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

definiert werden. Die Klasse eines Dokumentes enthält Vereinbarungen über logische Strukturen, z. B. die Gliederungseinheiten (Kapitel etc.), Listen und Umgebungen, die für alle Dokumente einer Klasse gemeinsam sind.

Zwischen den geschwungenen Klammern *muß* eine der in Tabelle 1 angeführten Document Classes angegeben werden.

Tabelle 1: Document Classes

article	für Artikel in wissenschaftlichen Zeitschriften, Vorträge, Praktikumsarbeiten, Seminararbeiten, kürzere Berichte, Anträge, Gutachten, Programmbeschreibungen, Einladungen u. v. a.
report	für längere Berichte, die aus mehreren Kapiteln bestehen, Diplomarbeiten, Dissertationen, Skripten u. ä.
book	für Bücher
proc	für Konferenzbände (Proceedings)
letter	für Briefe (im <i>Local Guide</i> [3] ist angegeben, ob Varianten für spezielle Briefköpfe zur Verfügung stehen.)
slides	für Folien. Diese Document Class ersetzt das alte $\text{SLI}\text{T}\text{E}\text{X}$ -Format.

Zwischen den eckigen Klammern *können*, durch Kommata getrennt, eine oder mehrere Optionen für Varianten der Standard-Layouts angegeben werden. Die wichtigsten Optionen sind in der Tabelle 2 angeführt.

Das Eingabefile für diese Beschreibung beginnt z. B. mit

```
\documentclass[11pt,a4paper,twoside]{article}
```

1.4.2 Pakete

Mit dem Befehl

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

können zusätzliche Pakete (packages) geladen werden. Eine Auswahl von Paketen findet sich in der Tabelle 3.

Tabelle 2: Document Class Options

<code>10pt</code>	für 10 Punkt hohe Schrift. Dies ist die Voreinstellung.
<code>11pt</code>	für 11 Punkte hohe Schrift, wie in dieser Beschreibung.
<code>12pt</code>	für 12 Punkt hohe Schrift, das ist etwa die bei Schreibmaschinen übliche Größe. Wenn keine dieser Optionen angegeben wird, verwendet \LaTeX 10 Punkte hohe Schrift, was auch eine beim Buchdruck übliche Größe ist.
<code>a4paper</code>	für Papier im DIN A4-Format. Ohne Angabe dieser Option nimmt $\LaTeX 2_{\epsilon}$ amerikanisches Papierformat an.
<code>fleqn</code>	für linksbündige statt zentrierte mathematische Gleichungen
<code>leqno</code>	für Gleichungsnummern links statt rechts von jeder Gleichung
<code>titlepage</code>	für eine eigene Titelseite (nur bei der Document Class <code>article</code>)
<code>twocolumn</code>	für zweispaltigen Druck
<code>twoside</code>	für Ausgabe mit unterschiedlichen rechten und linken Seiten.

1.4.3 Page Style

Das Seitenformat *kann* mit dem Befehl

```
\pagestyle{style}
```

festgelegt werden:

Wird kein `\pagestyle` angegeben oder `plain`, steht die Seitennummer in der Fußzeile. Bei `headings` stehen Kapitel-Überschrift und Seitennummer in der Kopfzeile. Bei `empty` sind Kopf- und Fußzeile leer.

Im *\LaTeX -Manual* [1] ist angegeben, wie man das Seiten-Layout und die Seitennumerierung außerdem mit den Befehlen `\thispagestyle`, `\pagenumbering`, `\twocolumn` und `\onecolumn` beeinflussen kann.

Tabelle 3: Packages (eine Auswahl)

<code>a4</code>	Anpassung an das A4-Papierformat, die über die Option <code>a4paper</code> hinausgeht.
<code>array</code>	Erweiterte Versionen der Umgebungen <code>array</code> , <code>tabular</code> und <code>tabular*</code> .
<code>babel</code>	Anpassungen für viele verschiedene Sprachen. Die gewählten Sprachen werden als Optionen angegeben.
<code>float</code>	Neue Optionen für Gleitobjekte (Tabellen, Abbildungen etc.), wie genauere Kontrolle über die Platzierung, Angabe von Stilparametern.
<code>ftnright</code>	Plaziert bei zweispaltigem Layout Fußnoten am unteren Ende der rechten Spalte.
<code>german</code>	Anpassungen für die deutsche Sprache, wie die Eingabe von Umlauten, deutsche Texte in Überschriften, deutsche Trennung, <code>\frenchspacing</code> .
<code>graphics</code>	Einbindung von Graphiken und Farbe (falls möglich) für verschiedene Drucker. Dieses Paket stellt umfangreiche Möglichkeiten zur Verfügung, die in [5] beschrieben sind.
<code>indentfirst</code>	Einzug auch beim ersten Absatz eines Kapitels.
<code>latexsym</code>	Stellt die L ^A T _E X-Symbole wie \square zur Verfügung.
<code>longtable</code>	Tabellen über mehrere Seiten, die automatisch umbrochen werden.
<code>makeidx</code>	Befehle zur automatischen Erstellung eines Index.
<code>multicol</code>	Mehrspaltiger Satz mit Kolumnenausgleich.
<code>newlfont</code>	Definiert die Befehle <code>\bf</code> , <code>\it</code> etc. so, daß sich ihre Wirkung kombiniert.
<code>showkeys</code>	Druckt die Namen aller <code>\labels</code> , <code>\refs</code> und <code>\pagerefs</code> im Text aus.
<code>supertabular</code>	Tabellen über mehrere Seiten mit automatischem Umbruch.
<code>theorem</code>	Erweiterung der <code>theorem</code> -Umgebung.
<code>verbatim</code>	Flexible Erweiterung der <code>verbatim</code> -Umgebung.

2 Setzen von Text

2.1 Zeilen- und Seiten-Umbruch

2.1.1 Blocksatz

Normaler Text wird im Blocksatz, d. h. mit Randausgleich gesetzt. \LaTeX führt den Zeilen- und Seitenumbruch automatisch durch. Dabei wird für jeden Absatz die bestmögliche Aufteilung der Wörter auf die Zeilen bestimmt, und – wenn notwendig – werden Wörter automatisch abgeteilt.

Das Ende von Wörtern und Sätzen wird durch Leerzeichen gekennzeichnet. Hierbei spielt es keine Rolle, ob man ein oder 100 Leerzeichen eingibt.

Eine oder mehrere Leerzeilen kennzeichnen das Ende von Absätzen.

Das Ende von Wörtern und Sätzen wird durch Leerzeichen gekennzeichnet.

Hierbei spielt es keine Rolle, ob man ein oder 100 Leerzeichen eingibt.

Eine oder mehrere Leerzeilen kennzeichnen das Ende von Absätzen.

Wie die Absätze gesetzt werden, hängt von der Document Class ab. In Artikeln, Berichten und Büchern werden Absätze durch das Einrücken der ersten Zeile dargestellt – so wie in allen anderen Kapiteln dieser Beschreibung. In Briefen oder wenn man Definitionen der Form

```
\setlength{\parindent}{0pt}
\setlength{\parskip}{5pt plus 2pt minus 1pt}
```

angibt, werden Absätze durch vertikalen Abstand ohne horizontales Einrücken dargestellt – so wie hier in diesem Abschnitt. Als Nebeneffekt werden dabei auch die Abstände bei den Umgebungen und bei mathematischen Gleichungen verändert.

Mit Hilfe der in Abschnitt 2.8 beschriebenen Umgebungen ist es möglich, spezielle Textteile jeweils anders zu setzen.

Für Ausnahmefälle kann man den Umbruch außerdem mit den folgenden Befehlen beeinflussen: Der Befehl `\` oder `\newline` bewirkt einen Zeilenwechsel ohne neuen Absatz, der Befehl `*` einen Zeilenwechsel, bei dem kein Seitenwechsel erfolgen darf. Der Befehl `\newpage` bewirkt einen Seitenwechsel. Mit den Befehlen `\linebreak[n]`, `\nolinebreak[n]`, `\pagebreak[n]` und `\nopagebreak[n]` kann man angeben, ob an bestimmten Stellen ein Zeilen- bzw. Seitenwechsel eher günstig oder eher ungünstig ist, wobei n die Stärke der Beeinflussung angibt (1, 2 oder 3).

Mit dem \LaTeX 2_ϵ -Befehl `\enlargethispage{Länge}` läßt sich eine gegebene Seite um einen festen Betrag verlängern oder verkürzen. Damit ist es möglich, noch eine Zeile mehr auf eine Seite zu bekommen.

\LaTeX bemüht sich, den Zeilenumbruch besonders schön zu machen. Falls es keine den strengen Regeln genügende Möglichkeit für einen glatten rechten Rand findet, läßt es eine Zeile zu lang und gibt eine entsprechende Fehlermeldung aus (“overfull hbox”). Dies tritt insbesondere dann auf, wenn es keine geeignete Stelle für die Silbentrennung findet. Innerhalb der `sloppypar`-Umgebung ist \LaTeX generell weniger streng in seinen Ansprüchen und vermeidet solche überlange Zeilen, indem es die Wortabstände stärker – notfalls auch unschön – vergrößert. In diesem Fall werden zwar Warnungen gemeldet (“underfull hbox”), das Ergebnis ist aber meistens durchaus brauchbar.

2.1.2 Silbentrennung

Falls die automatische Silbentrennung in einzelnen Fällen nicht das richtige Ergebnis liefert, kann man diese Ausnahmen mit den folgenden Befehlen richtigstellen. Dies kann insbesondere bei zusammengesetzten oder fremdsprachigen Wörtern notwendig werden.

Der Befehl `\hyphenation` bewirkt, daß die darin angeführten Wörter jedesmal an den und nur an den mit - markierten Stellen abgeteilt werden können. Er sollte am Beginn des Eingabefiles stehen und eignet sich *nur* für Wörter, die keine Umlaute, scharfes s, Ziffern oder sonstige Sonderzeichen enthalten¹. Beispiel:

```
\hyphenation{ Eingabe-file
             Eingabe-files FORTRAN }
```

Der Befehl `\-` innerhalb eines Wortes bewirkt, daß dieses Wort dieses eine Mal an den und nur an den mit `\-` markierten Stellen abgeteilt werden kann. Dieser Befehl eignet sich für *alle* Wörter, auch für solche, die Umlaute, scharfes s, Ziffern oder sonstige Sonderzeichen enthalten.

Mit dem Paket `german` steht eine bessere Möglichkeit zur Verfügung, nämlich der Befehl `"-`. Dieser erlaubt auch die Trennung an nicht explizit angegebenen Stellen im Wort (vgl. Abschnitt 2.4).

Eingabefile, \LaTeX -Eingabe-	Ein\ -gabe\ -file,
file, Häßlichkeit	\LaTeX-Eingabe\ -file,
	H"a"s\ -lich\ -keit

Für Spezialfälle gibt es auch noch den Befehl `\discretionary`. Die ersten beiden Parameter geben die Textteile für den Fall der Abteilung an, der dritte den Text ohne Abteilung.²

Drucker oder Druk-	Dru\discretionary{k-}{k}{ck}er
ker	

Der Befehl `\mbox{...}` bewirkt, daß das Argument nicht abgeteilt werden kann.

¹Bei der Benutzung des Paketes `T1enc` können auch die Umlaute und das scharfe s in `\hyphenation` verwendet werden.

²Eine bequemere Möglichkeit für die Angabe von „ck“ wird in Abschnitt 2.4 angegeben.

2.2.4 Ligaturen

Im Buchdruck ist es üblich, manche Buchstabenkombinationen anders zu setzen als die Einzelbuchstaben.

ff fi fl AV Te ... statt ff fi fl AV Te ...

Diese Ligaturen können vermieden werden, indem man zwischen die Buchstaben den Befehl `\/` setzt.³ Um die Lesbarkeit des Textes zu verbessern, sollte dies immer dann gemacht werden, wenn solche Buchstabenkombinationen nach Vorsilben oder bei zusammengesetzten Wörtern zwischen den Wortteilen auftreten.

Nicht Auflage (Au-fl-age)	Nicht Auflage (Au-fl-age) <code>\/</code>
sondern Auflage (Auf-lage)	sondern Auf <code>\/</code> lage (Auf-lage)

Mit dem Paket `german` steht zusätzlich der Befehl `"|` zur Verfügung, der gleichzeitig eine Trennhilfe darstellt.

Auflage (Auf-lage)	Auf <code>" </code> lage (Auf-lage)
--------------------	-------------------------------------

2.2.5 Akzente und spezielle Buchstaben

\LaTeX ermöglicht die Verwendung von Akzenten und speziellen Buchstaben aus zahlreichen verschiedenen Sprachen (siehe Tabelle 4). Akzente werden darin jeweils am Beispiel des Buchstabens `o` gezeigt, können aber prinzipiell auf jeden Buchstaben gesetzt werden. Wenn ein Akzent auf ein `i` oder `j` gesetzt werden soll, muß der `i`-Punkt wegbleiben. Dies erreicht man mit den Befehlen `\i` und `\j`.

Hôtel, naïve, smørebrød.	<code>H\^otel, na\i ve,</code>
¡Señorita!	<code>sm\o rebr\o d. \</code>
	<code>!‘Se\~norita!</code>

2.3 Abstände

Um einen glatten rechten Rand zu erreichen, variiert \LaTeX die Leerstellen zwischen den Wörtern etwas. Nach Punkten, Fragezeichen u. a., die einen Satz beenden, wird dabei ein etwas größerer Abstand erzeugt, was die Lesbarkeit des Textes erhöht. \LaTeX nimmt an, daß Punkte, die auf einen Großbuchstaben folgen, eine Abkürzung bedeuten, und daß alle anderen Punkte einen Satz beenden.

Außerdem gibt es die Möglichkeit, mit dem Befehl

`\frenchspacing`

³siehe auch Abschnitt 2.4.

Tabelle 4: Akzente und spezielle Buchstaben

Eingabe	Ausgabe	Eingabe	Ausgabe
<code>\‘o</code>	ò	<code>\’o</code>	ó
<code>\^o</code>	ô	<code>\~o</code>	õ
<code>\=o</code>	ō	<code>\.o</code>	ò
<code>\u o</code>	ö	<code>\v o</code>	õ
<code>\H o</code>	ő	<code>\"o</code>	ö
<code>\c o</code>	ç	<code>\d o</code>	ç
<code>\b o</code>	ç	<code>\r o</code>	ç
<code>\t oo</code>	ôô		
<code>\oe</code>	œ	<code>\OE</code>	Œ
<code>\ae</code>	æ	<code>\AE</code>	Æ
<code>\aa</code>	å	<code>\AA</code>	Å
<code>\o</code>	ø	<code>\O</code>	Ø
<code>\l</code>	ł	<code>\L</code>	Ł
<code>\i</code>	ı	<code>\j</code>	ĵ
<code>!‘</code>	ı	<code>?‘</code>	ı

zu vereinbaren, daß die Abstände an Satzenden nicht anders behandelt werden sollen als die zwischen Wörtern. Diese Konvention ist im nicht-englischen Sprachraum verbreitet. In diesem Fall brauchen die Befehle `\` und `\@` nicht angegeben werden. Mit dem Paket `german` ist `\frenchspacing` automatisch gewählt, dies kann durch

```
\nonfrenchspacing
```

wieder rückgängig gemacht werden.

Ausnahmen von diesen Regeln muß man `LATEX` mit den folgenden Befehlen mitteilen: Ein Backslash vor einem Leerzeichen bedeutet, daß diese Leerstelle nicht verbreitert werden darf. Eine `~` (Tilde) bedeutet eine Leerstelle, die nicht verbreitert werden darf und an der auch kein Zeilenwechsel erfolgen darf. Mit `\`, läßt sich ein kurzer Abstand erzeugen, wie er z. B. in Abkürzungen vorkommt. Der Befehl `\@` vor einem Punkt bedeutet, daß dieser Punkt einen Satz beendet, obwohl davor ein Großbuchstabe steht.

Dazu zählen u. a. auch die Österr.
Bundesbahnen.
Dr. Partl wohnt im 1. Stock.
... 5 m breit.
Ich brauche Vitamin C. Du nicht?

```
Dazu z"ahlen u.\,a.\ auch die
"Osterr.\ Bundesbahnen.  \
Dr.~Partl wohnt im 1.~Stock.  \
\dots\ 5~m breit.  \
Ich brauche Vitamin~C\@.
Du nicht?
```

2.4 Deutschsprachige Texte

Die in diesem Abschnitt angegebenen Befehle entsprechen der beim 6. Treffen der deutschen T_EX-Interessenten beschlossenen Norm für deutsche T_EX- und L^AT_EX-Befehle [10, 11]. Sie werden durch den Befehl

```
\usepackage{german}
```

aktiviert, in der aus Amerika stammenden Originalversion von L^AT_EX stehen sie nicht zur Verfügung. Im *Local Guide* [3] jeder L^AT_EX-Installation ist angegeben, welche der in diesem Abschnitt beschriebenen Befehle zur Verfügung stehen bzw. welche anderen Befehle eventuell an ihrer Stelle verwendet werden können.

Beim Austausch von L^AT_EX-Files mit anderen Installationen, die nicht denselben Befehlssatz verwenden, kann es also notwendig sein, spezielle im Schriftstück verwendete Pakete mitzusenden.

2.4.1 Silbentrennung

Beim Aufruf von L^AT_EX [3] wird festgelegt, ob die Silbentrennung für das gesamte Schriftstück nach englischen oder deutschen Regeln [8] erfolgen soll. Ausnahmen innerhalb des Schriftstücks können mit den in Abschnitt 2.1.2 angeführten Befehlen angegeben werden.

2.4.2 Umlaute und scharfes s

In der Originalversion von L^AT_EX gibt es den Befehl `\"` für Umlaute und `\ss` für scharfes s. Man kann deutsche Texte also in der etwas mühsamen folgenden Form schreiben:

Die häßliche Straße muß schöner werden.	Die h\"a\ss liche Stra\ss e mu\ss{} sch\"oner werden.
---	---

Wenn man das Paket `german` [9, 11] angibt, stehen zusätzlich die folgenden, einfacher verwendbaren Befehle zur Verfügung: Umlaute werden durch Voranstellen von Quotes geschrieben, also z. B. `"o` für „ö“. Für scharfes s schreibt man `"s` (ohne Probleme mit nachfolgenden Leerstellen). Man kann also schreiben:

Die häßliche Straße muß schöner werden.	Die h"a"sliche Stra"se mu"s sch"oner werden.
---	--

2.4.3 Anführungszeichen

„Deutsche Gänsefüßchen“ sehen anders aus als „englische Quotes“. In Original-L^AT_EX kann man versuchen, für deutsche Anführungszeichen unten (links) zwei Kommata und oben (rechts) zwei Grave-Akzente einzugeben, das Ergebnis ist aber nicht besonders schön. Statt `!‘‘` und `?‘‘` muß man `!\/'‘` bzw. `?\'/'‘` schreiben, weil man sonst die spanischen Sonderzeichen erhalten würde.

„Nein,“ sagte er, „ich weiß nichts!“	„,Nein,‘‘ sagte er, „,ich wei"s nichts!\/'‘“
--------------------------------------	--

Bei Benutzung des Paketes `german` [9, 11] stehen die folgenden Befehle für „richtige“ deutsche Anführungszeichen zur Verfügung: `"` (Quote und Grave-Akzent) für Anführungszeichen unten, und `'` (Quote und Apostroph) für Anführungszeichen oben.

```
„Nein,“ sagte er, „ich weiß nichts!“      "‘Nein,“’ sagte er,
                                           "‘ich wei"s nichts!“’
```

2.4.4 Sonstige Befehle

Das Paketes `german` macht einige weitere Befehle verfügbar. Die wichtigsten von ihnen sind: `"ck` für „ck“, das als „k-k“ abgeteilt wird, `"ff` für „ff“, das als „ff-f“ abgeteilt wird (und ebenso für andere Konsonanten), `"|` für die Vermeidung von Ligaturen und `"~` für einen Bindestrich, an dem kein Zeilenumbruch stattfinden soll.

Drucker bzw. Druk-ker	<code>Dru"cker \</code>
Rolladen bzw. Roll-laden	<code>Ro"lladen \</code>
Auflage	<code>Auf" lage \</code>
x-beliebig	<code>x"~beliebig \</code>
bergauf und -ab	<code>bergauf und "~ab</code>

2.4.5 Überschriften und Datumsangaben

In der Originalversion von \LaTeX sind die englischen Bezeichnungen für Kapitel, Abbildungen, Tabellen, Inhaltsverzeichnis usw. und für Datumsangaben voreingestellt. Dazu werden Befehlsnamen verwendet, die vom Benutzer undefiniert werden können [10, 11]. So bewirkt z. B. der Befehl

```
\renewcommand{\contentsname}{Inhalt}
```

daß die Überschrift des Inhaltsverzeichnisses aus dem Wort „Inhalt“ besteht. Diese Änderungen werden i. a. nicht direkt in jedem einzelnen \LaTeX -Eingabefile angegeben sondern indirekt durch die Angabe eines entsprechenden Paketes oder eines Befehls wie z. B.

```
\selectlanguage{\german}
```

Tabelle 5 enthält eine unvollständige Liste von Befehlsnamen, die in verschiedenen Document Classes verwendet werden, und der Texte, die sie in der englischen Originalversion und bei der Benutzung des Paketes `german` enthalten.

2.5 Kapitel und Überschriften

Der Beginn eines Kapitels bzw. Unterkapitels und seine Überschrift werden mit Befehlen der Form `\section{...}` angegeben. Dabei muß die logische Hierarchie eingehalten werden.

Bei Artikeln:

```
\section \subsection \subsubsection
```

Bei Berichten und Büchern:

```
\chapter \section \subsection \subsubsection
```

Artikel können also relativ einfach als Kapitel in ein Buch eingebaut werden. Die Abstände zwischen den Kapiteln, die Numerierung und die Schriftgröße der Überschrift werden von \LaTeX automatisch bestimmt.

Die Überschrift des gesamten Artikels bzw. die Titelseite des Schriftstücks wird mit dem Befehl `\maketitle` gesetzt. Der Inhalt muß vorher mit den Befehlen `\title`, `\author` und `\date` vereinbart werden (vgl. Abbildung 2 auf Seite 10).

Der Befehl `\tableofcontents` bewirkt, daß ein Inhaltsverzeichnis ausgedruckt wird. \LaTeX nimmt dafür immer die Überschriften und Seitennummern von der jeweils letzten vorherigen Verarbeitung des Eingabefiles. Bei einem neu erstellten oder um neue Kapitel erweiterten Schriftstück muß man das Programm \LaTeX also zweimal aufrufen, damit man die richtigen Angaben erhält.

Es gibt auch Befehle der Form `\section*{...}`, bei denen keine Numerierung und keine Eintragung ins Inhaltsverzeichnis erfolgen.

Mit den Befehlen `\label` und `\ref` ist es möglich, die von \LaTeX automatisch vergebenen Kapitelnummern im Text anzusprechen. Für `\ref{...}` setzt \LaTeX die mit `\label{...}` definierte Nummer ein. Auch hier wird immer die Nummer von der letzten vorherigen Verarbeitung des Eingabefiles genommen. Beispiel:

```
\section{Algorithmen}
```

```
...
```

```
Der Beweis daf"ur ist in Kapitel~\ref{bew} angegeben.
```

Tabelle 5: Überschriften

<code>\contentsname</code>	Contents	Inhaltsverzeichnis
<code>\listfigurename</code>	List of Figures	Abbildungsverzeichnis
<code>\listtablename</code>	List of Tables	Tabellenverzeichnis
<code>\abstractname</code>	Abstract	Zusammenfassung
<code>\refname</code>	References	Literatur
<code>\bibname</code>	Bibliography	Literaturverzeichnis
<code>\indexname</code>	Index	Index
<code>\figurename</code>	Figure	Abbildung
<code>\tablename</code>	Table	Tabelle
<code>\partname</code>	Part	Teil
<code>\chaptername</code>	Chapter	Kapitel
<code>\appendixname</code>	Appendix	Anhang
<code>\pagename</code>	Page	Seite

```
...
\section{Beweise} \label{bew}
...
```

2.6 Fußnoten

Fußnoten⁴ werden automatisch numeriert und am unteren Ende der Seite ausgedruckt.

```
Fußnoten\footnote
{Das ist eine Fußnote}
werden automatisch ...
```

2.7 Hervorgehobene Wörter

In maschinengeschriebenen Texten werden hervorzuhobende Texte unterstrichen, im Buchdruck werden dafür verschiedene Schriftarten verwendet. Der Befehl `\em` (*emphasize*) schaltet auf die „hervorstechende“ Schriftart um. Diese Schriftart bleibt bis zum Ende der aktuellen Gruppe eingeschaltet, der Befehl `\em` soll daher stets *innerhalb* von geschwungenen Klammern stehen.

Diese Klammer steht *vor* dem Befehl, *nicht nach* dem Befehl.

```
Diese Klammer steht {\em vor\/}
dem Befehl,
{\em nicht nach\/} dem Befehl.
```

\LaTeX verwendet für den hervorgehobenen Text *kursive* Schrift, in der alle Zeichen schräg nach rechts geneigt sind. Der Befehl `\/` ist notwendig, damit der letzte schräge Buchstabe nicht in den nachfolgenden geraden Text bzw. Abstand hineinragt.

Das *Nach*lager ist *nicht* brauchbar.
Das *Nacht*lager ist *nicht* häßlich.

```
Das {\em Nacht}lager
ist {\em nicht} brauchbar. \\
Das {\em Nacht\/}lager
ist {\em nicht\/} h"a"slich.
```

Werden *innerhalb eines hervorgehobenen Textes* nochmals *Wörter hervorgehoben*, so nimmt \LaTeX dafür eine *aufrechte Schrift*.

\LaTeX 2_ε kann in den meisten Fällen die Korrektur automatisch durchführen, wozu der Befehl

```
\emph{hervorgehobener Text}
```

benutzt wird. Sollte die Korrektur nicht erwünscht sein (etwa vor einem Punkt oder Komma), so ist

```
\emph{hervorgehobener Text\nocorr}
```

zu benutzen. Dabei muß `\nocorr` unmittelbar vor der schließenden Klammer stehen.

⁴Das ist eine Fußnote

2.8 Umgebungen

Die Kennzeichnung von speziellen Textteilen, die anders als im normalen Blocksatz gesetzt werden sollen, erfolgt mittels sogenannter Umgebungen (environments) in der Form

```
\begin{name} text \end{name}
```

Umgebungen sind *Gruppen*. Sie können auch ineinander geschachtelt werden, dabei muß aber die richtige Reihenfolge beachtet werden:

```
\begin{aaa}...\begin{bbb}...\end{bbb}...\end{aaa}
```

2.8.1 Zitate (quote, quotation, verse)

Die `quote`-Umgebung eignet sich für kürzere Zitate, hervorgehobene Sätze und Beispiele. Der Text wird links und rechts eingerückt:

Eine typographische Faustregel für die Zeilenlänge lautet:

Keine Zeile soll mehr als 66 Buchstaben enthalten.

Deswegen werden in Zeitungen mehrere Spalten nebeneinander verwendet.

Eine typographische Faustregel für die Zeilenlänge lautet:

```
\begin{quote}
Keine Zeile soll mehr als
66~Buchstaben enthalten.
\end{quote}
```

Deswegen werden in Zeitungen mehrere Spalten nebeneinander verwendet.

Die `quotation`-Umgebung unterscheidet sich in den Standardklassen von der `quote`-Umgebung dadurch, daß Absätze durch Einzüge gekennzeichnet werden. Sie ist daher für längere Zitate, die aus mehreren Absätzen bestehen, geeignet.

Die `verse`-Umgebung eignet sich für Gedichte und für Beispiele, bei denen die Zeilenaufteilung wesentlich ist. Die Verse (Zeilen) werden durch `\\` getrennt, Strophen durch Leerzeilen.

2.8.2 Listen (itemize, enumerate, description)

Die Umgebung `itemize` eignet sich für einfache Listen (siehe Abbildung 3). Die Umgebung `enumerate` eignet sich für nummerierte Aufzählungen (siehe Abbildung 4). Die Umgebung `description` eignet sich für Beschreibungen (siehe Abbildung 5).

2.8.3 Linksbündig, rechtsbündig, zentriert (flushleft, flushright, center)

Die Umgebungen `flushleft` und `flushright` bewirken links- bzw. rechtsbündigen Satz („Flattersatz“, d. h. ohne Randausgleich), `center` setzt den Text in die Mitte der Zeile. Die einzelnen Zeilen werden durch `\\` getrennt. Wenn man `\\` nicht angibt, bestimmt \LaTeX automatisch die Zeilenaufteilung (siehe Abbildung 6 auf Seite 25).

<p>Listen:</p> <ul style="list-style-type: none"> • Bei <code>itemize</code> werden die Elemente durch Punkte und andere Symbole gekennzeichnet. • Listen können auch geschachtelt werden: <ul style="list-style-type: none"> – Die maximale Schachtelungstiefe ist 4. – Einrückung und Bezeichnung der Elemente wechseln automatisch. • usw. 	<pre> Listen: \begin{itemize} \item Bei {\tt itemize} werden die Elemente ... \item Listen können auch geschachtelt werden: \begin{itemize} \item Die maximale ... \item Einrückung und ... \end{itemize} \item usw. \end{itemize} </pre>
---	---

Abbildung 3: Beispiel für `itemize`

<p>Aufzählungen:</p> <ol style="list-style-type: none"> 1. Bei <code>enumerate</code> werden die Elemente mit Ziffern oder Buchstaben numeriert. 2. Die Numerierung erfolgt automatisch. 3. Listen können auch geschachtelt werden: <ol style="list-style-type: none"> (a) Die maximale Schachtelungstiefe ist 4. (b) Einrückung und Bezeichnung der Elemente wechseln automatisch. 4. usw. 	<pre> Aufzählungen: \begin{enumerate} \item Bei {\tt enumerate} werden die Elemente ... \item Die Numerierung ... \item Listen können auch geschachtelt werden: \begin{enumerate} \item Die maximale ... \item Einrückung und ... \end{enumerate} \item usw. \end{enumerate} </pre>
--	--

Abbildung 4: Beispiel für `enumerate`

Kleine Tierkunde:	Kleine Tierkunde:
Gelse: ein kleines Tier, das östlich des Semmering Touristen verjagt.	<code>\begin{description}</code>
	<code>\item[Gelse:]</code>
	ein kleines Tier, das ...
	<code>\item[Gemse:]</code>
	ein gro"ses Tier, das ...
Gemse: ein großes Tier, das westlich des Semmering von Touristen verjagt wird.	<code>\item[G"urteltier:]</code>
	ein mittelgro"ses Tier,
	das ...
Gürteltier: ein mittelgroßes Tier, das hier nur wegen der Länge seines Namens vorkommt.	<code>\end{description}</code>

Abbildung 5: Beispiel für `description`

links	<code>\begin{flushleft}</code>
Backbord	links \\
	Backbord
	<code>\end{flushleft}</code>
	<code>\begin{flushright}</code>
	rechts \\
rechts	Steuerbord
Steuerbord	
	<code>\end{flushright}</code>
	<code>\begin{center}</code>
Im	Im \\ Reich \\ der \\ Mitte
Reich	
der	
Mitte	<code>\end{center}</code>

Abbildung 6: Linksbündig, rechtsbündig und zentriert

2.8.4 Direkte Ausgabe (`verbatim`, `verb`)

Zwischen `\begin{verbatim}` und `\end{verbatim}` stehende Zeilen werden genauso ausgedruckt, wie sie eingegeben wurden, d. h. mit allen Leerzeichen und Zeilenwechslern und ohne Interpretation von Spezialzeichen und \LaTeX -Befehlen. Dies eignet sich z. B. für das Ausdrucken eines (kurzen) Computer-Programms.

Innerhalb eines Absatzes können einzelne Zeichenkombinationen oder kurze Textstücke ebenso „wörtlich“ ausgedruckt werden, indem man sie zwischen `\verb|` und `|` einschließt. Mit diesen Befehlen wurden z. B. alle \LaTeX -Befehle in der vorliegenden Beschreibung gesetzt.

Der `\dots`-Befehl ...

Der `\verb|\dots|`-Befehl `\dots`

Die `verbatim`-Umgebung und der Befehl `\verb` dürfen *nicht* innerhalb von Parametern von anderen Befehlen und auch nicht innerhalb der `tabular`-Umgebung verwendet werden.

2.8.5 Abbildungen (`figure`)

Zwischen `\begin{figure}` und `\end{figure}` stehender Text – bzw. der mit `\vspace` angegebene Platz für das Einkleben eines Bildes – wird automatisch an eine Stelle gesetzt, wo er komplett hinpaßt, ohne durch einen Seitenwechsel zerrissen zu werden. Mit `\caption{...}` setzt man die Bezeichnung der Abbildung. Dabei ist nur der Text anzugeben, das Wort „Abbildung“ und die fortlaufende Nummer werden von \LaTeX hinzugefügt. Bei Abbildungen ist es allgemein üblich, die Bezeichnung *unter* das Bild zu setzen. Mit `\label` und `\ref` kann man die Nummer der Abbildung im Text ansprechen.

Abbildung 7 zeigt ein Beispiel aus der Pop-Art.

```
Abbildung~\ref{weiss} zeigt ein
Beispiel aus der Pop-Art.
\begin{figure}[htbp]
\vspace{6cm}
\caption{R.~Black, Ganz
in wei"s} \label{weiss}
\end{figure}
```

Die Parameter in den eckigen Klammern dienen zur Platzierung der Abbildung. Sie werden von \LaTeX stets in der Reihenfolge ausgewertet: `h` ‘here’ (hier), `t` ‘top’ (oben auf der Seite), `b` ‘bottom’ (unten auf der Seite) oder `p` ‘page’ (eigene Seite für Abbildungen). Durch Weglassen z. B. von `t` läßt sich eine Abbildung nach unten schieben. Die Voreinstellung ist `tbp` ohne `h`. Eine Abbildung, die nicht plazierte werden konnte, wird von \LaTeX immer weiter nach hinten verschoben (und schiebt alle weiteren Abbildungen vor sich her!), bis ein neues Kapitel beginnt, das Dokument zu Ende ist, oder der Befehl

```
\clearpage
```

eingegeben wird.

Abbildung 7: R. Black, Ganz in weiß

$\text{\LaTeX} 2_{\epsilon}$ kennt einen weiteren Platzierungsparameter, ! ‘bang’, der vorübergehend alle Platzierungsbeschränkungen aufhebt. Bang muß immer zusammen mit mindestens einem der 4 anderen Parameter benutzt werden.

Der Befehl `\label` muß nach dem `\caption`-Befehl stehen, sonst stimmt die Numerierung nicht.

2.8.6 Tabellen (table)

Tabellen werden analog zu Abbildungen zwischen `\begin{table}` und `\end{table}` gesetzt, `\caption`, `\label` und `\ref` wirken analog. Bei Abbildungen sind beide möglichen Konventionen verbreitet: Die Bezeichnung wird entweder immer *über* oder immer *unter* die Tabelle gesetzt.

Für das Zusammensetzen der Tabelle wird meistens die `tabbing`- oder die `tabular`-Umgebung verwendet, die im folgenden beschrieben werden. Man kann aber auch andere Strukturen (z.B. `enumerate` oder `description`) verwenden. Abbildung 8 enthält eine Skizze für eine solche Schachtelung von Umgebungen.

```

\begin{table}[htbp]
\caption{...} \label{...}
  \begin{center}
    \begin{tabular}{...}
      ...
    \end{tabular}
  \end{center}
\end{table}

```

Abbildung 8: Aufbau einer `table`-Umgebung

2.8.7 Tabulatoren (tabbing)

In der `tabbing`-Umgebung kann man Tabulatoren ähnlich wie an Schreibmaschinen setzen und verwenden. Der Befehl `\=` setzt eine Tabulatorposition, `\kill` bedeutet, daß die „Musterzeile“ nicht ausgedruckt werden soll, `\>` springt zur nächsten Tabulatorposition, und `\\` trennt die Zeilen.

links	Mittelteil	rechts	<code>\begin{tabbing}</code>
Es			<code>war einmal\quad \=</code>
war einmal	und ist	nicht mehr	<code>Mittelteil\quad \= \kill</code>
ein		ausgestopfter	<code>links \> Mittelteil \> rechts\\</code>
		Teddybär	<code>Es \\</code>
			<code>war einmal \> und ist</code>
			<code>\> nicht mehr\\</code>
			<code>ein \> \> ausgestopfter\\</code>
			<code>\> \> Teddyb"ar</code>
			<code>\end{tabbing}</code>

2.8.8 Eigentliche Tabellen (tabular)

Die `tabular`-Umgebung dient zum Setzen von Tabellen, bei denen \LaTeX automatisch die benötigte Spaltenbreite bestimmt, und bei der auch spezielle Eigenschaften wie Rechtsbündigkeit und Hilfslinien vereinbart werden können.

Im Parameter des Befehls `\begin{tabular}{...}` wird das Format der Tabelle angegeben. Dabei bedeutet `l` eine Spalte mit linksbündigem Text, `r` eine mit rechtsbündigem, `c` eine mit zentriertem Text, `p{breite}` eine Spalte der angegebenen Breite mit mehrzeiligem Text, `|` einen senkrechten Strich.

Innerhalb der Tabelle bedeutet `&` den Sprung in die nächste Tabellenspalte, `\\` trennt die Zeilen, `\hline` (an Stelle einer Zeile) setzt einen waagrechten Strich.

7C0	hexadezimal
3700	oktal
11111000000	binär
1984	dezimal

```

\begin{tabular}{|rl|}
\hline
7C0 & hexadezimal \\
3700 & oktal \\
11111000000 & bin"ar \\
\hline\hline
1984 & dezimal \\
\hline
\end{tabular}

```

3 Setzen von mathematischen Formeln

3.1 Allgemeines

Mathematische Textteile innerhalb eines Absatzes werden zwischen `\(` und `\)` oder zwischen `$` und `$` oder zwischen `\begin{math}` und `\end{math}` eingeschlossen. Als mathematische Texte gelten sowohl komplette mathematische Formeln als auch einzelne Variablennamen, die sich auf Formeln beziehen, griechische Buchstaben, das Hoch- und Tiefstellen von Texten und diverse Sonderzeichen.

Seien a und b die Katheten und c die Hypotenuse, dann gilt $c^2 = a^2 + b^2$ (Pythagoräischer Lehrsatz).

Seien a und b die Katheten und c die Hypotenuse, dann gilt $c^2 = a^2 + b^2$ (Pythagoraischer Lehrsatz).

\TeX spricht man wie $\tau\epsilon\chi$ aus.

100 m² Nutzfläche

Mit ♡-lichen Grüßen

`\TeX` spricht man wie
 `$\tau\epsilon\chi$` aus.
 `100~m^{2}$` Nutzfl"ache
 Mit `\heartsuit`-lichen
 Gr"u"sen

Größere mathematische Formeln oder Gleichungen setzt man besser in eigene Zeilen. Dazu werden sie zwischen `\begin{displaymath}` und `\end{displaymath}` oder zwischen `[` und `]` gesetzt, wenn sie keine Gleichungsnummer erhalten sollen, bzw. zwischen `\begin{equation}` und `\end{equation}`, wenn sie eine Gleichungsnummer erhalten sollen.

Seien a und b die Katheten und c die Hypotenuse, dann gilt

$$c = \sqrt{a^2 + b^2} \quad (1)$$

(Pythagoräischer Lehrsatz).

Seien a und b die Katheten und c die Hypotenuse, dann gilt
`\begin{equation}`
`c = \sqrt{ a^2+b^2 }`
`\end{equation}`
 (Pythagoraischer Lehrsatz).

Mit `\label` und `\ref` kann man die Gleichungsnummern im Text ansprechen.

$$\epsilon > 0 \quad (2)$$

Aus (2) folgt ...

`\begin{equation} \label{eps}`
`\epsilon > 0`
`\end{equation}`

Aus `(\ref{eps})` folgt `\dots`

Das Setzen im mathematischen Modus unterscheidet sich vom Text-Modus vor allem durch folgende Punkte:

1. Leerstellen und Zeilenwechsel haben bei der Eingabe keine Bedeutung, alle Abstände werden nach der Logik der mathematischen Ausdrücke automatisch bestimmt oder müssen durch spezielle Befehle wie `\,` oder `\quad` angegeben werden.

$$\forall x \in \mathbf{R} : \quad x^2 \geq 0 \quad (3)$$

```

\begin{equation}
\forall x \in \mathbf{R} :
\quad x^2 \geq 0
\end{equation}

```

2. Leerzeilen sind verboten (Mathematischen Formeln müssen innerhalb eines Absatzes stehen).
3. Jeder einzelne Buchstabe wird als Name einer Variablen betrachtet und entsprechend gesetzt (kursiv mit zusätzlichem Abstand). Will man innerhalb eines mathematischen Textes normalen Text (in aufrechter Schrift, mit Wortabständen) setzen, muß man diesen in `\textrm{...}` einschließen.

$$x^2 \geq 0 \quad \text{für alle } x \in \mathbf{R} \quad (4)$$

```

\begin{equation}
x^2 \geq 0 \quad \text{für alle } x \in \mathbf{R}
\end{equation}

```

3.2 Elemente in mathematischen Formeln

In diesem Abschnitt werden die wichtigsten Elemente, die in mathematischen Formeln verwendet werden, kurz beschrieben. Eine Liste aller verfügbaren Symbole enthält Kapitel 3.5.

Kleine **griechische Buchstaben** werden als `\alpha`, `\beta`, `\gamma`, usw. eingegeben, große griechische Buchstaben als `\mathrm{A}`, `\mathrm{B}`, `\Gamma`, `\Delta`, usw.

$$\lambda, \xi, \pi, \mu, \Phi, \Omega$$

```

$\lambda, \xi, \pi, \mu, \Phi, \Omega$

```

Weiters gibt es eine Fülle von **mathematischen Symbolen**: von \in über \Rightarrow bis ∞ (siehe Kapitel 3.5).

Mathematische Symbole können in $\text{L}^{\text{T}}\text{E}^{\text{X}} 2_{\epsilon}$ aus einer Auswahl von **Alphabeten** genommen werden.

ABCabc	<code> \$\mathrm{ABCabc}\$ \</code>
ABCabc	<code> \$\mathbf{ABCabc}\$ \</code>
ABCabc	<code> \$\mathsf{ABCabc}\$ \</code>
ABCabc	<code> \$\mathhtt{ABCabc}\$ \</code>
<i>ABC</i>	<code> \$\mathcal{ABC}\$</code>

Die kalligraphischen Buchstaben (`\mathcal`) gibt es nur als Großbuchstaben. Lokal können weitere Alphabete, z. B. Fraktur, zur Verfügung stehen (siehe *Local Guide* [3]).

Exponenten und Indizes können mit den Zeichen `^` und `_` hoch- bzw. tiefgestellt werden.

a_1	x^2	$e^{-\alpha t}$	a_{ij}^3	<code>\\$a_{1}\\$ \quad</code>
				<code>\\$x^{2}\\$ \quad</code>
				<code>\\$e^{-\alpha t}\\$ \quad</code>
				<code>\\$a^{3}_{ij}\\$</code>

Das **Wurzelzeichen** wird mit `\sqrt` eingegeben, n -te Wurzeln mit `\sqrt[n]`. Die Größe des Wurzelzeichens wird von L^AT_EX automatisch gewählt.

\sqrt{x}	$\sqrt{x^2 + \sqrt{y}}$	$\sqrt[3]{2}$	<code>\\$\sqrt{x}\\$ \quad</code>
			<code>\\$\sqrt{x^2+\sqrt{y}}\\$</code>
			<code>\quad \\$\sqrt[3]{2}\\$</code>

Die Befehle `\overline` und `\underline` bewirken **waagrechte Striche** direkt über bzw. unter einem Ausdruck.

$\overline{m+n}$	<code>\\$\overline{m+n}\\$</code>
------------------	-----------------------------------

Die Befehle `\overbrace` und `\underbrace` bewirken **waagrechte Klammern** über bzw. unter einem Ausdruck.

$\underbrace{a+b+\cdots+z}_{26}$	<code>\\$\underbrace{a+b+\cdots+z}_{26}\\$</code>
----------------------------------	---

Um mathematische **Akzente** wie Pfeile oder Schlangen auf Variablen zu setzen, gibt es die in Tabelle 6 angeführten Befehle. Längere Tilden und Dacherln, die sich über mehrere (bis zu 3) Zeichen erstrecken können, erhält man mit `\widetilde` bzw. `\widehat`. Ableitungszeichen werden mit `'` (Apostroph) eingegeben.

$y = x^2$	$y' = 2x$	$y'' = 2$	<code>\begin{displaymath}</code>
			<code>y=x^{2} \quad</code>
			<code>y'=2x \quad</code>
			<code>y''=2</code>
			<code>\end{displaymath}</code>

Mathematische **Funktionen** werden in der Literatur üblicherweise nicht kursiv (wie die Namen von Variablen), sondern in „normaler“ Schrift dargestellt. Dazu gibt es die folgenden Befehle:

<code>\arccos</code>	<code>\cos</code>	<code>\csc</code>	<code>\exp</code>	<code>\ker</code>	<code>\limsup</code>	<code>\min</code>	<code>\sinh</code>
<code>\arcsin</code>	<code>\cosh</code>	<code>\deg</code>	<code>\gcd</code>	<code>\lg</code>	<code>\ln</code>	<code>\Pr</code>	<code>\sup</code>
<code>\arctan</code>	<code>\cot</code>	<code>\det</code>	<code>\hom</code>	<code>\lim</code>	<code>\log</code>	<code>\sec</code>	<code>\tan</code>
<code>\arg</code>	<code>\coth</code>	<code>\dim</code>	<code>\inf</code>	<code>\liminf</code>	<code>\max</code>	<code>\sin</code>	<code>\tanh</code>

Für die Modulo-Funktion gibt es zwei verschiedene Befehle: `\bmod` für den binären Operator $a \bmod b$ und `\pmod{\dots}` für die Angabe in der Form $x \equiv a \pmod{b}$.

$$\lim_{x \rightarrow 0} \frac{\sin x}{x} = 1$$

```

\begin{displaymath}
\lim_{x \to 0} \frac{\sin x}{x} = 1
\end{displaymath}

```

Ein **Bruch** (fraction) wird mit dem Befehl `\frac{\dots}{\dots}` gesetzt. Für einfache Brüche kann man aber auch den Operator `/` verwenden.

$$1\frac{1}{2} \text{ Stunden} \quad \frac{x^2}{k+1} \quad x^{\frac{2}{k+1}} \quad x^{1/2}$$

```

$1\frac{1}{2}$~Stunden
\begin{displaymath}
\frac{x^2}{k+1} \quad x^{\frac{2}{k+1}} \quad x^{1/2}
\end{displaymath}

```

Binomial-Koeffizienten können in der Form `\dots\choose\dots` gesetzt werden. Mit dem Befehl `\atop` erhält man das Gleiche ohne Klammern.

$$\binom{n}{k} \quad x \atop y+2$$

```

\begin{displaymath}
{n \choose k} \quad x \atop y+2
\end{displaymath}

```

Das **Integralzeichen** wird mit `\int` eingegeben, das **Summenzeichen** mit `\sum`. Die obere und untere Grenze wird mit `^` bzw. `_` wie beim Hoch/Tiefstellen angegeben.

Normalerweise werden die Grenzen neben das Integralzeichen gesetzt (um Platz zu sparen), durch Einfügen des Befehl `\limits` wird erreicht, daß die Grenzen oberhalb und unterhalb des Integralzeichens gesetzt werden.

Beim Summenzeichen hingegen werden die Grenzen bei der Angabe von `\nolimits` oder im laufenden Text neben das Summenzeichen gesetzt, ansonsten aber unter- und oberhalb.

$$\sum_{i=1}^n \quad \int_0^{\frac{\pi}{2}} \quad \int_{-\infty}^{+\infty}$$

```

\begin{displaymath}
\sum_{i=1}^n \quad \int_0^{\frac{\pi}{2}} \quad \int_{-\infty}^{+\infty}
\end{displaymath}

```

Für **Klammern** und andere Begrenzer gibt es in \TeX viele verschiedene Symbole (z. B. `[` `<` `||` `\Downarrow`). Runde und eckige Klammern können mit den entsprechenden Tasten eingegeben werden, geschwungene mit `\{`, die anderen mit speziellen Befehlen (z. B. `\updownarrow`).

Setzt man den Befehl `\left` vor öffnende Klammern und den Befehl `\right` vor schließende, so wird automatisch die richtige Größe gewählt.

$$1 + \left(\frac{1}{1-x^2} \right)^3$$

```
\begin{displaymath}
1 + \left( \frac{1}{1-x^2} \right)^3
\end{displaymath}
```

In manchen Fällen möchte man die Größe der Klammern lieber selbst festlegen, dazu sind die Befehle `\bigl`, `\Bigl`, `\biggl` und `\Biggl` anstelle von `\left` und analog `\bigr` etc. anstelle von `\right` anzugeben.

$$\left((x+1)(x-1) \right)^2$$

```
\begin{displaymath}
\Bigl( (x+1) (x-1) \Bigr)^2
\end{displaymath}
```

Um in Formeln **3 Punkte** (z. B. für $1, 2, \dots, n$) auszugeben, gibt es die Befehle `\ldots` und `\cdots`. `\ldots` setzt die Punkte auf die Grundlinie (low), `\cdots` setzt sie in die Mitte der Zeilenhöhe (centered). Außerdem gibt es die Befehle `\vdots` für vertikale und `\ddots` für diagonale Punkte.

$$x_1, \dots, x_n \quad x_1 + \cdots + x_n$$

```
\begin{displaymath}
x_{1}, \ldots, x_{n} \quad \text{\quad}
x_{1} + \cdots + x_{n}
\end{displaymath}
```

3.3 Nebeneinander Setzen

Wenn man mit den von TEX gewählten **Abständen** innerhalb von Formeln nicht zufrieden ist, kann man sie mit expliziten Befehlen verändern. Die wichtigsten sind `\,` für einen sehr kleinen Abstand, `_` für einen mittleren, `\quad` und `\qquad` für große Abstände sowie `\!` für die Verkleinerung eines Abstands.

$$F_n = F_{n-1} + F_{n-2} \quad n \geq 2$$

```
\begin{displaymath}
F_{n} = F_{n-1} + F_{n-2}
\quad n \geq 2
\end{displaymath}
```

$$\iint_D dx dy \quad \text{statt} \quad \int \int_D dx dy$$

```
\begin{displaymath}
\int\!\!\!\int\!\!\!\int_D dx\,dy
\quad \text{\texttrm{statt}} \quad
\int\int_D dx dy
\end{displaymath}
```

3.4 Übereinander Setzen

Für **Matrizen** u. ä. gibt es die `array`-Umgebung, die ähnlich wie die `tabular`-Umgebung funktioniert. Der Befehl `\!` trennt die Zeilen.

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots \\ x_{21} & x_{22} & \dots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

```
\begin{displaymath}
\mathbf{X} =
\left( \begin{array}{ccc}
x_{11} & x_{12} & \dots \\
x_{21} & x_{22} & \dots \\
\vdots & \vdots & \ddots
\end{array} \right)
\end{displaymath}
```

Für **mehrzeilige** Formeln oder Gleichungssysteme verwendet man die Umgebungen `eqnarray` und `eqnarray*` statt `equation`. Bei `eqnarray` erhält jede Zeile eine eigene Gleichungsnummer, bei `eqnarray*` wird wie bei `displaymath` keine Nummer hinzugefügt. Für Gleichungssysteme, die *eine* gemeinsame Nummer erhalten sollen, kann man eine `array`-Umgebung innerhalb der `equation`-Umgebung verwenden.

Die Umgebungen `eqnarray` und `eqnarray*` funktionieren wie eine 3-spaltige Tabelle der Form `{rc1}`, wobei die mittlere Spalte für das Gleichheits- oder Ungleichheitszeichen verwendet wird, nach dem die Zeilen ausgerichtet werden sollen. Der Befehl `\` trennt die Zeilen.

$f(x) = \cos x$	(5)	$f'(x) = -\sin x$	(6)
$\int_0^x f(y)dy = \sin x$	(7)		

```
\begin{eqnarray}
f(x) & = & \cos x & \\
f'(x) & = & -\sin x & \\
\int_0^x f(y)dy & = & \sin x & \\
\end{eqnarray}
```

Zu lange Gleichungen werden von L^AT_EX *nicht* automatisch abgeteilt. Der Autor muß bestimmen, an welcher Stelle abgeteilt und wie weit eingerückt werden soll. Meistens verwendet man dafür eine der beiden folgenden Varianten:

$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$	(8)		

```
\begin{eqnarray}
\sin x & = & x - \frac{x^3}{3!} & + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots \\
\end{eqnarray}
```

$\cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots$	(9)		

```
\begin{eqnarray}
\left[ \cos x = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots \right]
\end{eqnarray}
```

Der Befehl `\nonumber` bewirkt, daß an diese Stelle keine Gleichungsnummer gesetzt wird. Der Befehl `\lefteqn` ermöglicht Ausnahmen von der Spaltenaufteilung innerhalb `eqnarray`. Genauere Informationen enthält das *L^AT_EX-*Manual** [1].

3.5 Liste der mathematischen Symbole

In den folgenden Tabellen sind alle Symbole angeführt, die standardmäßig im mathematischen Modus verwendet werden können. Die mit einem * versehenen Symbole sind in L^AT_EX_{2 ϵ} nur mit dem Paket `latexsym` verwendbar. Bei vielen Installationen stehen mit den Paketen `amssymb` und `wasysmb` weitere Zeichen zur Verfügung, näheres steht im *Local Guide* [3].

Tabelle 6: Mathematische Akzente

<code>\hat a</code>	\hat{a}	<code>\check a</code>	\check{a}
<code>\tilde a</code>	\tilde{a}	<code>\acute a</code>	\acute{a}
<code>\grave a</code>	\grave{a}	<code>\dot a</code>	\dot{a}
<code>\ddot a</code>	\ddot{a}	<code>\breve a</code>	\breve{a}
<code>\bar a</code>	\bar{a}	<code>\vec a</code>	\vec{a}

Tabelle 7: Große griechische Buchstaben

Γ	<code>\Gamma</code>	Ξ	<code>\Xi</code>	Φ	<code>\Phi</code>
Δ	<code>\Delta</code>	Π	<code>\Pi</code>	Ψ	<code>\Psi</code>
Θ	<code>\Theta</code>	Σ	<code>\Sigma</code>	Ω	<code>\Omega</code>
Λ	<code>\Lambda</code>	Υ	<code>\Upsilon</code>		

Tabelle 8: Kleine griechische Buchstaben

α	<code>\alpha</code>	ι	<code>\iota</code>	ϱ	<code>\varrho</code>
β	<code>\beta</code>	κ	<code>\kappa</code>	σ	<code>\sigma</code>
γ	<code>\gamma</code>	λ	<code>\lambda</code>	ς	<code>\varsigma</code>
δ	<code>\delta</code>	μ	<code>\mu</code>	τ	<code>\tau</code>
ϵ	<code>\epsilon</code>	ν	<code>\nu</code>	υ	<code>\upsilon</code>
ε	<code>\varepsilon</code>	ξ	<code>\xi</code>	ϕ	<code>\phi</code>
ζ	<code>\zeta</code>	\omicron	<code>\omicron</code>	φ	<code>\varphi</code>
η	<code>\eta</code>	π	<code>\pi</code>	χ	<code>\chi</code>
θ	<code>\theta</code>	ϖ	<code>\varpi</code>	ψ	<code>\psi</code>
ϑ	<code>\vartheta</code>	ρ	<code>\rho</code>	ω	<code>\omega</code>

Tabelle 9: Verschiedene sonstige Symbole

\aleph	<code>\aleph</code>	$'$	<code>\prime</code>	\forall	<code>\forall</code>
\hbar	<code>\hbar</code>	\emptyset	<code>\emptyset</code>	\exists	<code>\exists</code>
\imath	<code>\imath</code>	∇	<code>\nabla</code>	\neg	<code>\neg</code>
\jmath	<code>\jmath</code>	\surd	<code>\surd</code>	\flat	<code>\flat</code>
ℓ	<code>\ell</code>	\top	<code>\top</code>	\natural	<code>\natural</code>
\wp	<code>\wp</code>	\perp	<code>\perp</code>	\sharp	<code>\sharp</code>
\Re	<code>\Re</code>	\parallel	<code>\parallel</code>	\clubsuit	<code>\clubsuit</code>
\Im	<code>\Im</code>	\sphericalangle	<code>\sphericalangle</code>	\diamondsuit	<code>\diamondsuit</code>
∂	<code>\partial</code>	\triangle	<code>\triangle</code>	\heartsuit	<code>\heartsuit</code>
∞	<code>\infty</code>	\backslash	<code>\backslash</code>	\spadesuit	<code>\spadesuit</code>
\mho	<code>\mho</code>	\square	<code>\square</code>	\diamond	<code>\diamond</code>

Tabelle 10: „Große“ Operatoren

\sum	<code>\sum</code>	\bigcap	<code>\bigcap</code>	\bigodot	<code>\bigodot</code>
\prod	<code>\prod</code>	\bigcup	<code>\bigcup</code>	\bigotimes	<code>\bigotimes</code>
\coprod	<code>\coprod</code>	\bigsqcup	<code>\bigsqcup</code>	\bigoplus	<code>\bigoplus</code>
\int	<code>\int</code>	\bigvee	<code>\bigvee</code>	\biguplus	<code>\biguplus</code>
\oint	<code>\oint</code>	\bigwedge	<code>\bigwedge</code>		

Tabelle 11: Binäre Operatoren

+	$+$	-	$-$		
\pm	<code>\pm</code>	\cap	<code>\cap</code>	\vee	<code>\vee</code>
\mp	<code>\mp</code>	\cup	<code>\cup</code>	\wedge	<code>\wedge</code>
\setminus	<code>\setminus</code>	\oplus	<code>\oplus</code>	\oplus	<code>\oplus</code>
\cdot	<code>\cdot</code>	\sqcap	<code>\sqcap</code>	\ominus	<code>\ominus</code>
\times	<code>\times</code>	\sqcup	<code>\sqcup</code>	\otimes	<code>\otimes</code>
$*$	<code>\ast</code>	\triangleleft	<code>\triangleleft</code>	\oslash	<code>\oslash</code>
\star	<code>\star</code>	\triangleright	<code>\triangleright</code>	\odot	<code>\odot</code>
\diamond	<code>\diamond</code>	\wr	<code>\wr</code>	\dagger	<code>\dagger</code>
\circ	<code>\circ</code>	\bigcirc	<code>\bigcirc</code>	\ddagger	<code>\ddagger</code>
\bullet	<code>\bullet</code>	\triangleup	<code>\triangleup</code>	\amalg	<code>\amalg</code>
\div	<code>\div</code>	∇	<code>\nabla</code>		

Tabelle 12: Relationen

$<$	<code><</code>	$>$	<code>></code>	$=$	<code>=</code>
\leq	<code>\leq</code>	\geq	<code>\geq</code>	\equiv	<code>\equiv</code>
\prec	<code>\prec</code>	\succ	<code>\succ</code>	\sim	<code>\sim</code>
\preceq	<code>\preceq</code>	\succeq	<code>\succeq</code>	\simeq	<code>\simeq</code>
\ll	<code>\ll</code>	\gg	<code>\gg</code>	\asymp	<code>\asymp</code>
\subset	<code>\subset</code>	\supset	<code>\supset</code>	\approx	<code>\approx</code>
\subseteq	<code>\subseteq</code>	\supseteq	<code>\supseteq</code>	\cong	<code>\cong</code>
\sqsubset	<code>\sqsubset</code>	\sqsupseteq	<code>\sqsupseteq</code>	\bowtie	<code>\bowtie</code>
\in	<code>\in</code>	\ni	<code>\ni</code>	\Join^*	<code>\Join^*</code>
\vdash	<code>\vdash</code>	\dashv	<code>\dashv</code>	\models	<code>\models</code>
\smile	<code>\smile</code>	\mid	<code>\mid</code>	\doteq	<code>\doteq</code>
\frown	<code>\frown</code>	\parallel	<code>\parallel</code>	\perp	<code>\perp</code>
\sqsubset^*	<code>\sqsubset^*</code>	\sqsupset^*	<code>\sqsupset^*</code>	\propto	<code>\propto</code>
\lhd^*	<code>\lhd^*</code>	\unlhd^*	<code>\unlhd^*</code>	\rhd^*	<code>\rhd^*</code>
		\unrhd^*	<code>\unrhd^*</code>		

Tabelle 13: Negationen

\nless	<code>\not<</code>	\nlessgtr	<code>\not></code>	\neq	<code>\not=</code>
\nlessoreq	<code>\not\leq</code>	\ngtr	<code>\not\geq</code>	\nequiv	<code>\not\equiv</code>
\nprec	<code>\not\prec</code>	\nsucc	<code>\not\succ</code>	\nsim	<code>\not\sim</code>
\npreceq	<code>\not\preceq</code>	\nsucceq	<code>\not\succeq</code>	\nsimeq	<code>\not\simeq</code>
\nsubset	<code>\not\subset</code>	\nsupset	<code>\not\supset</code>	\napprox	<code>\not\approx</code>
\nsubseteq	<code>\not\subseteq</code>	\nsupseteq	<code>\not\supseteq</code>	\ncong	<code>\not\cong</code>
\nsubsetseq	<code>\not\subsetseq</code>	\nsupseteq	<code>\not\supseteq</code>	\nasymp	<code>\not\asymp</code>

Tabelle 14: Pfeile

\leftarrow	<code>\leftarrow</code>	\longleftarrow	<code>\longleftarrow</code>	\uparrow	<code>\uparrow</code>
\Lleftarrow	<code>\Lleftarrow</code>	\Longleftarrow	<code>\Longleftarrow</code>	\Uparrow	<code>\Uparrow</code>
\rightarrow	<code>\rightarrow</code>	\longrightarrow	<code>\longrightarrow</code>	\downarrow	<code>\downarrow</code>
\Rrightarrow	<code>\Rrightarrow</code>	\Longrightarrow	<code>\Longrightarrow</code>	\Downarrow	<code>\Downarrow</code>
\leftrightarrow	<code>\leftrightarrow</code>	\longleftrightarrow	<code>\longleftrightarrow</code>	\updownarrow	<code>\updownarrow</code>
\Leftrightarrow	<code>\Leftrightarrow</code>	\Longleftrightarrow	<code>\Longleftrightarrow</code>	\Updownarrow	<code>\Updownarrow</code>
\mapsto	<code>\mapsto</code>	\longmapsto	<code>\longmapsto</code>	\nearrow	<code>\nearrow</code>
\hookleftarrow	<code>\hookleftarrow</code>	\hookrightarrow	<code>\hookrightarrow</code>	\searrow	<code>\searrow</code>
\leftharpoonup	<code>\leftharpoonup</code>	\rightharpoonup	<code>\rightharpoonup</code>	\swarrow	<code>\swarrow</code>
\leftharpoondown	<code>\leftharpoondown</code>	\rightharpoondown	<code>\rightharpoondown</code>	\nwarrow	<code>\nwarrow</code>
\rightleftharpoons	<code>\rightleftharpoons</code>			\leadsto	<code>\leadsto*</code>

Tabelle 15: Linke Klammern

$($	<code>(</code>	$[$	<code>[</code>	$\{$	<code>\{</code>
\lbrack	<code>\lbrack</code>	\lfloor	<code>\lfloor</code>	\lceil	<code>\lceil</code>
\lbrace	<code>\lbrace</code>	\langle	<code>\langle</code>		

Tabelle 16: Rechte Klammern

$)$	<code>)</code>	$]$	<code>]</code>	$\}$	<code>\}</code>
\rbrack	<code>\rbrack</code>	\rfloor	<code>\rfloor</code>	\rceil	<code>\rceil</code>
\rbrace	<code>\rbrace</code>	\rangle	<code>\rangle</code>		

Tabelle 17: Synonyme

Für manche Symbole stehen mehrere verschiedene Befehle zur Verfügung.

\neq	<code>\ne</code> or <code>\neq</code>	<code>\not=</code>
\leq	<code>\le</code>	<code>\leq</code>
\geq	<code>\ge</code>	<code>\geq</code>
$\{$	<code>\{</code>	<code>\lbrace</code>
$\}$	<code>\}</code>	<code>\rbrace</code>
\rightarrow	<code>\to</code>	<code>\rightarrow</code>
\leftarrow	<code>\gets</code>	<code>\leftarrow</code>
\ni	<code>\owns</code>	<code>\ni</code>
\wedge	<code>\land</code>	<code>\wedge</code>
\vee	<code>\lor</code>	<code>\vee</code>
\neg	<code>\lnot</code>	<code>\neg</code>
$ $	<code>\vert</code>	<code> </code>
$\ $	<code>\Vert</code>	<code>\ </code>

Tabelle 18: Nicht-mathematische Symbole

Die folgenden Symbole sind im Text-Modus verfügbar:

\dagger	<code>\dag</code>	\S	<code>\S</code>	\copyright	<code>\copyright</code>
\ddagger	<code>\ddag</code>	\P	<code>\P</code>	\pounds	<code>\pounds</code>

4 Spezialitäten

Das komplette Menü der Spezialitäten, die von L^AT_EX serviert werden, ist im *L^AT_EX-Manual* [1] beschrieben. Hier soll nur auf einige besondere „Zuckerln“ hingewiesen werden.

4.1 Schriftarten und -größen (Fonts)

Normalerweise wählt L^AT_EX die geeigneten Schriftarten und Schriftgrößen auf Grund der Befehle aus, die die logische Struktur des Textes angeben (Überschriften, emphasize usw.). In Spezialfällen kann die Schriftart und -größe auch explizit mit den in den Tabellen 19 und 20 angeführten Befehlen gewechselt werden.

Die Verwendung dieser Befehle erfolgt analog zum Befehl `\em` innerhalb von Gruppen, für kurze Textstücke sollen die Formen `\text{...}`, die ein Argument haben, benutzt werden.

Die kleinen fetten Römer beherrschten das ganze große <i>Italien</i> .	<code>{\small Die kleinen \textbf{fetten}</code>
le 2 ^{ème} régime	<code>R"omer beherrschten }{\large das ganze gro"se \textit{Italien}. \\ le \$2^{\text{rm}\scriptsize\`eme}\$ r\`egime</code>

Die Größen-Befehle verändern auch die Zeilenabstände auf die jeweils dazupassenden Werte – aber nur, wenn die Leerzeile, die den Absatz beendet, innerhalb des Gültigkeitsbereichs des Größen-Befehls liegt, die schließende geschwungene Klammer darf also nicht zu früh kommen!

Je *weniger* verschiedene Schriftarten man verwendet, desto lesbarer und schöner wird das Schriftstück.

Tabelle 19: Schriftarten

<code>\textrm</code>	<code>\rm</code>	normale Schrift (roman)
<code>\textsf</code>	<code>\sf</code>	„sans serif“ Schrift
<code>\texttt</code>	<code>\tt</code>	Schreibmaschinenschrift (typewriter)
<code>\textbf</code>	<code>\bf</code>	fette Schrift (boldface)
<code>\textit</code>	<code>\it</code>	<i>kursive Schrift (italic)</i>
<code>\textsl</code>	<code>\sl</code>	<i>schräge Schrift (slanted)</i>
<code>\textsc</code>	<code>\sc</code>	KAPITÄLCHEN (CAPS AND SMALL CAPS)

Tabelle 20: Schriftgrößen

<code>\tiny</code>	winzig kleine Schrift
<code>\scriptsize</code>	sehr kleine Schrift (wie Indizes)
<code>\footnotesize</code>	kleine Schrift (wie Fußnoten)
<code>\small</code>	kleine Schrift
<code>\normalsize</code>	normale Schrift
<code>\large</code>	große Schrift
<code>\Large</code>	größere Schrift
<code>\LARGE</code>	sehr große Schrift
<code>\huge</code>	riesig groß
<code>\Huge</code>	gigantisch

4.2 Abstände

4.2.1 Zeilenabstand

Um in einem Schriftstück größere Zeilenabstände, als in der Document Class vorgesehen ist, zu verwenden, gibt es in $\text{\LaTeX} 2_{\epsilon}$ den Befehl `\linespread`:

für „eineinhalbzeilige“ Ausgabe:

```
\linespread{1.3}
```

für „doppelzeilige“ Ausgabe:

```
\linespread{1.6}
```

4.2.2 Spezielle horizontale Abstände

Die Abstände zwischen Wörtern und Sätzen werden von \LaTeX automatisch gesetzt. Sonstige horizontale Abstände kann man mit dem Befehl

```
\hspace{länge}
```

bewirken. Wenn der Abstand auch am Beginn oder Ende einer Zeile erhalten bleiben soll, muß `\hspace*` statt `\hspace` geschrieben werden. Die Längenangabe besteht im einfachsten Fall aus einer Zahl und einer Einheit. Die wichtigsten Einheiten sind in Tabelle 21 angeführt.

Hier ist 1.5 cm Abstand. Hier `\hspace{1.5cm}` ist 1.5~cm
Abstand.

Die Befehle in Tabelle 22 sind Abkürzungen für spezielle horizontale Abstände. Der Befehl `\hfill` kann dazu dienen, einen vorgegebenen Platz auszufüllen. Beispiele für die Verwendung von `\,` und `\hfill`:

„ ‚Parsifal‘ dauert länger als ‚Cats‘.“ "‘`\,` ‚Parsifal‘ dauert 1"anger
als ‚Cats‘."’

Verbindung mit `\pagebreak[4]` kann dazu dienen, Text an den unteren Rand einer Seite zu setzen oder vertikal zu zentrieren.

Zusätzliche Abstände zwischen zwei Zeilen *innerhalb* eines Absatzes oder einer Tabelle erreicht man mit dem Befehl

```
\\[länge]
```

4.3 Briefe (letter)

Wenn man als Document Class `letter` (oder eine Variante davon, siehe [3]) angibt, kann man zwischen `\begin{document}` und `\end{document}` einen oder mehrere Briefe schreiben.

Mit `\signature` und `\address` definiert man Name und Adresse des Absenders. `\begin{letter}{...}` beginnt einen Brief an den im Parameter mit Name und Adresse angegebenen Empfänger. `\opening{...}` und `\closing{...}` schreiben die Anrede und den abschließenden Gruß, an den automatisch die mit `\signature` vereinbarte Unterschrift angefügt wird. `\end{letter}` beendet den jeweiligen Brief.

Wenn man vor `\begin{document}` den Befehl `\makelabels` angibt, werden außerdem Adreß-Etiketten erzeugt.

Abbildung 9 enthält ein Beispiel für einen Brief.

```

\documentclass[12pt,a4paper]{letter}
\usepackage{german}
\address{EDV-Zentrum der TU Wien \\
         Abt.~Digitalrechenanlage \\
         Wiedner Hauptstra"se 8--10 \\ A-1040 Wien }
\signature{Dr.~Hubert Partl}
\begin{document}
\begin{letter}{Frau Mag.~Elisabeth Schlegl \\
             EDV-Zentrum der Karl-Franzens-Universit"at \\
             Attemsgasse 25/II \\ A-8010 Graz}
\opening{Liebe Frau Schlegl,}
herzlichen Dank f"ur die Zusendung .....

..... in etwa 2--3~Wochen fertig zu sein.
\closing{Mit freundlichen Gr"u"sen}
\end{letter}
\end{document}

```

Abbildung 9: Brief von H. P. an E. S.

4.4 Literaturangaben

Mit der `thebibliography`-Umgebung kann man ein Literaturverzeichnis drucken. Darin beginnt jede Literaturangabe mit `\bibitem`. Als Parameter wird

ein Name vereinbart, unter dem die Literaturstelle im Text mit `\cite` zitiert werden kann, und dann folgt der Text der Literaturangabe. Die Numerierung erfolgt automatisch. Der Parameter bei `\begin{thebibliography}` gibt die maximale Breite dieser Nummernangabe an, also z. B. `{99}` für maximal zweistellige Nummern.

Das Zitieren der Literaturstelle im Text erfolgt jeweils mit dem Befehl `\cite`. Beispiel:

Partl [1] hat vorgeschlagen, daß ...

```
Partl~\cite{pa} hat
vorgeschlagen ...
```

Literatur

[1] H. Partl: *German T_EX*, TUGboat Vol. 9, No. 1 (1988)

```
\begin{thebibliography}{99}
\bibitem{pa}
H.~Partl: {\it German \TeX,}
TUGboat Vol.~9, No.~1 (1988)
\end{thebibliography}
```

4.5 Robuste und zerbrechliche Befehle

Die meisten \LaTeX -Befehle sind „robust“, d. h. sie liefern immer das gewünschte Ergebnis.

Es gibt aber auch sogenannte „zerbrechliche“ Befehle, die in bestimmten Situationen (innerhalb von sogenannten „bewegten“ Parametern) nur dann richtig funktionieren, wenn man den Befehl `\protect` voranstellt. Zu den zerbrechlichen Befehlen zählen unter anderem die in Tabelle 20 auf Seite 41 angeführten Befehle, die die Schriftgröße verändern, und der Befehl `\footnote`. Es gibt also einige wenige (und sehr selten auftretende) Spezialfälle, in denen man z. B. `\protect\small` statt `\small` schreiben muß. Wann solche Spezialfälle auftreten, ist im *\LaTeX-Manual* [1] angegeben.

4.6 Kompatibilität zu $\text{\LaTeX}2.09$

Fast alle mit der alten Version $\text{\LaTeX}2.09$ erstellten Dokumente lassen sich mit $\text{\LaTeX}2_{\epsilon}$ fehlerfrei übersetzen. An dem Befehl `\documentstyle` erkennt $\text{\LaTeX}2_{\epsilon}$ die alte Version und verwendet dann einen speziellen Kompatibilitätsmodus, in dem die Erweiterungen abgeschaltet sind.

In $\text{\LaTeX}2.09$ geschriebene Dokumente lassen sich meistens leicht nach $\text{\LaTeX}2_{\epsilon}$ portieren. Hierzu ist die Zeile

```
\documentstyle[optionen]{stil}
```

durch die entsprechende Angabe

```
\documentclass[optionen]{klasse}
```

zu ersetzen. Dabei ist zu beachten, daß die meisten „Stiloptionen“ von $\text{\LaTeX}2.09$ zu Paketen in $\text{\LaTeX}2_{\epsilon}$ werden, die mit

```
\usepackage{pakete}
```

geladen werden. Die meisten Stiloptionen von $\text{\LaTeX}2.09$ lassen sich unverändert mit $\text{\LaTeX}2_{\epsilon}$ als Pakete weiterverwenden. Einige alte Optionen funktionieren nur dann als Pakete, wenn vorher das Paket `rawfonts` geladen wurde.

Ferner sollten alle Schriftwechsel, insbesondere in Formeln, von `{\rm ...}` auf `\textrm{...}` bzw. `\mathrm{...}` umgestellt werden.

Literatur

- [1] L. Lamport: *L^AT_EX, A Document Preparation System, User's Guide and Reference Manual*, Addison-Wesley Publishing Company, second edition (1994).
- [2] M. Goossens, F. Mittelbach und A. Samarin, *The L^AT_EX Companion*, Addison Wesley Publishing Company, (1994).
- [3] An jeder Installation (Rechenzentrum o. ä.) sollte ein *L^AT_EX Local Guide* erhältlich sein, in dem alle für die Installation spezifischen Angaben – z. B. die für den Aufruf der Programme notwendigen Befehle und die zur Verfügung stehenden Files, Document Styles und Fonts – angeführt sind.
- [4] An jeder L^AT_EX 2_ε-Installation müssen die folgenden 3 Guides vorhanden sein: *L^AT_EX 2_ε for authors* (`usrguide.tex`), *L^AT_EX 2_ε for class package writers* (`clsguide.tex`) und *L^AT_EX 2_ε font selection* (`fntguide.tex`).
- [5] D. P. Carlisle: *Packages in the 'graphics' Bundle* (`grfguide.tex`), Bestandteil des `graphics`-Bündels (1994). In diesem Guide sind die Konzepte und die Bedienung des `graphics`-Paketes erklärt.
- [6] D. E. Knuth: *The T_EXbook*, Band A der Reihe *Computers and Typesetting*, Addison-Wesley Publishing Company (1984).
- [7] N. Schwarz: *Einführung in T_EX*, Addison-Wesley Deutschland (1987).
- [8] B. Raichle und N. Schwarz: `ghyph31.tex` – optimierte deutsche Trenntabelle (1994) verfügbar am Server `ftp.dante.de` (Heidelberg).
- [9] B. Raichle (Koordinator): `german.sty` – Package zur Anpassung von L^AT_EX an die deutsche Sprache, verfügbar am Server `ftp.dante.de` (Heidelberg).
- [10] H. Partl: *Ein „Minimal Subset“ für einheitliche deutsche T_EX-Befehle*, Vortrag und Diskussion beim 6. Treffen der deutschen T_EX-Interessenten in Münster (1987, nicht veröffentlicht).
- [11] H. Partl: *German T_EX*, TUGboat Vol. 9, No. 1 (1988).
- [12] H. Partl: *Layout-Änderungen mit L^AT_EX 2.09*, EDV-Zentrum der Technischen Universität Wien (1988). Die L^AT_EX-Eingabefiles sind am Server `ftp.dante.de` (Heidelberg) verfügbar.