

Manuskript zur \LaTeX -Einführung

Wolfgang Hackl

wolf.hackl@gmx.at

<http://hackl.homeip.net/LaTeX-WS/LaTeX-Manuskript.pdf>

29. März 2005

\LaTeX ist ein programmierbares Textsatzsystem, das dafür entwickelt wurde, “schöne” Bücher zu machen. Mit \LaTeX gesetzte Dokumente sind von Natur aus geeignet, sofort zur Druckerei oder zu einem Verlag geschickt zu werden. Typischerweise wird \LaTeX von Leuten eingesetzt, die Publikationen im technischen und naturwissenschaftlichen Umfeld veröffentlichen. Der Clou an \LaTeX ist, daß sich der Autor wenig mit Layout-Vorgaben beschäftigen muß und sich somit auf den Inhalt des Dokuments konzentrieren kann. Das klingt einfach – \LaTeX ist aber anspruchsvoll zu bedienen. In diesem Dokument werden grundlegende Funktionalitäten von \LaTeX vorgestellt. Es wird erklärt, was man braucht, um einfache Texte zu setzen.



Inhaltsverzeichnis

Inhaltsverzeichnis	3
1. Ein Rezept für Ungeduldige	4
2. Grundlagen	4
2.1. Über dieses Dokument	4
2.2. Was ist L ^A T _E X?	4
2.3. Vor- und Nachteile	5
2.4. Bestandteile eines L ^A T _E X-Systems	6
2.4.1. Editor	7
2.4.2. Die L ^A T _E X-Umgebung	7
2.4.3. Viewer	7
2.4.4. Graphikbearbeitungstools	7
2.5. Generierungslauf	7
3. Syntax und Aufbau	9
3.1. Präambel	9
3.1.1. Dokumentklassen	9
3.1.2. Packages	11
3.1.3. Beispiel	11
3.2. Dokumentrumpf	13
3.3. Gliederungen	14
3.4. Ein Minimaldokument	15
4. Schriftbild	16
4.1. Schriftarten	16
4.2. Schriftgrößen	17
5. Querverweise	17
5.1. Fußnoten	17
5.2. Referenzen	17
6. Nützliche Umgebungen	18
6.1. Aufzählungen	18
6.2. Floats	19
6.2.1. Bilder	19
6.2.2. Tabellen	20
6.3. Verbatim	21
6.4. Mathematische Formeln	21
7. Verzeichnisse	23
A. Linksammlung	24

Bibliographie	25
----------------------	-----------

Abbildungsverzeichnis

1. Löwe	20
-------------------	----

1. Ein Rezept für Ungeduldige

- Texteditor starten und mit `\documentclass[a4paper]{article}` den Dokumenttyp angeben.
- Präambel schreiben:
 - `\usepackage[T1]{fontenc}` erweiterte Zeichensätze
 - `\usepackage{lmodern}` Skalierbare PostScript Fonts
 - `\usepackage[latin1]{inputenc}` Umlaute ermöglichen
 - Dokumenttitel in `\title{Titel}` und Autor in `\author{Autor}` angeben
- Mit `\begin{document}` den Dokumentrumpf beginnen
- Mittels `\maketitle` den Titel generieren lassen
- Abschnitte z. B. mit `\section{Überschrift}` und `\subsection{Unterüberschrift}` gliedern
- Text schreiben
- Mit `\end{document}` das Dokument beenden
- Mit `latex <dokument.tex>` oder `pdflatex <dokument.tex>` den Quellcode in ein dvi-Dokument oder PDF-Dokument übersetzen

2. Grundlagen

2.1. Über dieses Dokument

Dieses Manuskript entstand als Handout für Präsentationen bei der VALUG und der LUGL. Am 27. Jänner 2005 wurde es nach einem Dist-Upgrade von Debian Woody auf Sarge mit einer neueren Version von L^AT_EX übersetzt. Durch geänderte Parameternamen im Paket “listings” mußte der Sourcecode bei eingebundenen Listings angepaßt werden. Der Quellcode läßt sich mit älteren Versionen des Packages “listings” nicht mehr ohne Änderungen übersetzen (dafür können die anderen das Dokument ohne Änderungen übersetzen). Die geänderten Parameter findet man in der Dokumentation vom Paket “listings” in der Datei listings.dvi.

2.2. Was ist L^AT_EX?

T_EX ist ein programmierbares und mit Makros erweiterbares Textsatzsystem, das von Donald Knuth¹ 1978 entwickelt wurde. T_EX ist dafür gedacht, “schöne” Bücher zu ma-

¹Donald Knuths Homepage: <http://www-cs-faculty.stanford.edu/~knuth/>

chen - insbesondere Bücher, die sehr viel mathematischen Inhalt haben².

Leslie Lamport schuf 1985 mit L^AT_EX eine um viele Makropakete erweiterte Version. Sie erleichterte die Handhabung wesentlich, da sich der Autor nicht mehr so viel um typographische Details kümmern muß. Als nächste Erweiterung schufen Rainer Schöpf und Frank Mittelbach 1993 L^AT_EX 2_ε. Anfang Februar 2005 wurde Version L^AT_EX3 freigegeben.

In der Praxis sieht das Arbeiten mit einem Textsatzsystem so aus: Man schreibt mit einem beliebigen Texteditor den Inhalt des Dokuments. Wie der Text im fertigen Dokument aussehen und angeordnet sein soll, muß man mit den Befehlen einer Beschreibungssprache direkt mit in den Text verwoben angeben. Nach dem Schreiben wird der Quelltext durch einen Compiler in ein optisch ansprechendes Format übersetzt. Während man bei einem Textverarbeitungssystem den Text im Dokument nahezu beliebig mit der Maus gestaltet und verziert, wird das Dokument beim Textsatzsystem beschrieben und in einem zweiten Schritt nach strengen Regeln generiert. Dort, wo ebenso strenge Vorgaben bestehen, bietet sich L^AT_EX förmlich an.

2.3. Vor- und Nachteile

Mit L^AT_EX gesetzte Dokumente sind von Natur aus geeignet, sofort zur Druckerei oder zu einem Verlag geschickt zu werden. In akademischen Kreisen ist es üblich, Artikel für einschlägige Journale und Verlage in L^AT_EX zu setzen. Der Clou an L^AT_EX ist, daß sich der Autor nicht mit Layout-Vorgaben beschäftigen muß und sich somit auf den Inhalt des Dokuments konzentrieren kann. Der einzige erforderliche Schritt ist das Einbinden eines vom Verlag vorgegebenen Style-Files in das Dokument.

Ein kleiner Seitenhieb: Word hat nicht umsonst lediglich den Namen Word verdient und nicht etwa Article oder Book. L^AT_EX ist im Gegensatz dazu sehr wohl geeignet, Dokumente der Größenordnung eines Buches zu verarbeiten. Wer schon einmal versucht hat, Dokumente mit mehr als 3 Seiten und klickbarem Inhaltsverzeichnis, Formeln, Grafiken, Verweisen, Hyperlinks, Zitaten, Bibliographie, etc. mit Word zu schreiben, ist ein gebranntes Kind. Man lernt zu schätzen, daß die Überschrift so aussieht, wie erwartet oder daß das Inhaltsverzeichnis mit dem Dokument übereinstimmt.

LaTeX bietet einige Vorteile für den Anwender. Je nach Anwendungsgebiet sind diese mehr oder weniger relevant. Die wichtigsten seien im folgenden kurz genannt:

Kostenlose Verfügbarkeit Für nahezu jedes Betriebssystem gibt es LaTeX als Download im Internet. Daraus ergibt sich

LaTeX ist plattformunabhängig und -übergreifend In LaTeX erstellte Dokumente können auf nahezu jedem Computersystem betrachtet, editiert und gedruckt werden. Der Quelltext ist reiner ASCII-Text, der einfach weitergegeben werden kann.

²UK T_EX FAQ <http://www.tex.ac.uk/cgi-bin/texfaq2html?label=whatTeX>

LaTeX liefert erstklassige typographische Ergebnisse Anders als übliche Textverarbeitungen erfüllen mit LaTeX gesetzte Dokumente selbst höchste Anforderungen ohne umfassende Kenntnisse seitens des Anwenders vorauszusetzen.

Mathematischer Formelsatz Speziell Techniker, Mathematiker und andere Anwender, die häufig Formeln in ihre Dokumente integrieren müssen, können mittels LaTeX einwandfreie Ergebnisse produzieren.

Logische statt physischer Formatierung Dies bedeutet, dass LaTeX die gesamte Formatierung übernimmt, während der Anwender nur die Struktur des Dokuments festlegt.

Stabilität LaTeX ist eines der stabilsten Programmpakete überhaupt, was sich auch daran zeigt, dass die aktuelle Version seit 1994 existiert, und seither nur marginale Änderungen erfahren hat. Datenverluste gibt es faktisch nicht.

Gibt es auch Nachteile? Wo Licht ist, ist auch immer Schatten. Die wesentlichen Nachteile sollen ebenso Erwähnung finden:

Steile Lernkurve Die ersten Schritte mit LaTeX sind recht kompliziert; der Einstieg in OpenOffice und andere WYSIWYG-Editoren fällt zu Beginn erheblich leichter. Jedoch folgt auf den - zugegeben recht anspruchsvollen - Einstieg ein wesentlich einfacheres Arbeiten als mit erwähnter Konkurrenz. Das Einfügen eines Inhaltsverzeichnis und vor allem dessen Aktualhalten ist mit LaTeX ungleich leichter als in Word.

Kein offizieller Support Da LaTeX nicht das Produkt einer Firma ist, kann man auch nirgendwo anrufen um Hilfe zu bekommen. Statt dessen ist man bei Fragen auf die Suche in zahlreich vorhandenen Büchern, Internetseiten, Mailinglisten und Anwendervereinigungen angewiesen.

Komplizierte Modifikation der Standardlayouts Wer eigene Vorlagen für Dokumente erstellen möchte muss tief in die Materie eintauchen. Zwar gibt es bereits zahlreiche Vorlagen, aber personalisierte Dokumente erfordern ein großes Vorwissen.

(Geklaut bei irgendeiner Web Site [[vorteil](#)]).

2.4. Bestandteile eines L^AT_EX-Systems

Die neben dem L^AT_EX-Quellcode notwendigen Komponenten für das Arbeiten mit L^AT_EX sind nachfolgend beschrieben.

2.4.1. Editor

Zum Setzen des \LaTeX -Codes genügt ein Texteditor nach Wahl. Hierbei reichen die Geschmäcker von "echten" Editoren wie vi mit Syntax Highlighting und Emacs mit Plugins zu integrierten Umgebungen wie Kile und LyX³.

Für Anwender, die Windows benutzen, gibt es MiKTeX⁴ als Implementierung des TeX-Systems und das TeXnicCenter⁵ als Editor. Aus unerklärlichen Gründen werden die Dokumente unter Windows jedoch nie so schön, wie unter einem unixartigen System...

2.4.2. Die \LaTeX -Umgebung

Natürlich benötigt man auch die \LaTeX -Umgebung selbst, um die Sourcen zu übersetzen. Dazu gehören neben dem Compiler hauptsächlich Informationen über Schriften und Formatvorlagen für unterschiedliche Dokumenttypen. Teil der Basisausstattung sind auch Werkzeuge, um die Bibliographie-Datenbank zu verwalten. Bei Debian ist man mit den Paketen `tetex-base`, `tetex-bin`, `tetex-extra` und natürlich `tetex-doc` für den Haus- und Hofgebrauch ausreichend ausgerüstet.

2.4.3. Viewer

Für die Betrachtung der generierten Dokumente benötigt man einen Viewer für die unterschiedlichen Ausgabeformate `.dvi`, `.ps` oder `.pdf`. DVI ist das Standardausgabeformat von \LaTeX . Beim Paket `tetex-bin` von Debian Woody ist der dafür gängigste Viewer `xdvi` enthalten. Wer die Dokumente mit `pdflatex` direkt in das PDF übersetzt ist mit `xpdf` als Viewer gut beraten. Dieser kann im Gegensatz zum Adobe Reader geänderte Dokumente neu laden.

2.4.4. Graphikbearbeitungstools

Zur Aufbereitung von Bildmaterial benötigt man diverse Bearbeitungs- und Umwandlungstools. Für nachträgliche Konvertierungen generierter Dokumente braucht man ebenfalls Werkzeuge. Die gängigsten sind `gimp`, `dvips` und `ps2pdf`

2.5. Generierungslauf

Als Input für einen Generierungslauf dienen in erster Linie die `.tex`-Quelldateien, die der Autor erstellt. Die eingebundenen Pakete bzw. Styles (`.sty`) und Dokumentklassen (`.cls`) müssen ebenfalls vorhanden sein, um das Dokument zu übersetzen. Bei Debian

³Hinweise zur Auswahl des Editors: <http://www.dante.de/faq/de-tex-faq/html/woher.html#25>

⁴MiKTeX-Homepage: <http://www.miktex.org>

⁵TeXnicCenter-Homepage: <http://www.texniccenter.org>

liegen die Dateien z. B. unter `/usr/share/texmf`. Achtung: Hat man zusätzliche `.sty`- oder `.cls`-Dateien von jemand anders bekommen, so muß man die Datenbank als `root` mit dem Befehl `texhash` aktualisieren.

Während eines \LaTeX -Generierungslaufes werden verschiedene Files erzeugt:

- `.aux` - Querverweise, etc
- `.toc` - erzeugt durch die Angabe von `\tableofcontents`.
- `.lof` - erzeugt durch die Angabe von `\listoffigures`.
- `.lot` - erzeugt durch die Angabe von `\listoftables`.
- `.log` - eine Kopie der Logging-Ausgaben, die am Schirm erscheinen

Dadurch, daß das Endprodukt intern in mehreren Schritten erzeugt wird, kann es im Fall eines Generierungsfehlers zu unerklärlichen Phänomenen kommen. Typischerweise ist dies so, wenn man Überschriften ändert. Liest man die Meldung

```
LaTeX Warning: Label(s) may have changed. Rerun to get  
cross-references right.,
```

so ist das Problem durch einen erneuten Generierungslauf zu beheben.

Wenn ein Fehler beim \LaTeX -Lauf aufgetreten ist, stehen folgende Befehle zur Verfügung:

<Eingabetaste> Diesen Fehler überspringen

h Hilfen zum Fehler

x Abbruch der Generierung

s Die weiteren Fehlermeldungen ohne anzuhalten ausgeben

r Übersetzung ohne Halt fortsetzen

q Übersetzung ohne Halt und Ausgaben fortsetzen

i Beliebigen Text (auch \LaTeX -Befehle) einfügen

e Einen Editor aufrufen und die \LaTeX -Datei editieren

1..9 die nächsten 1..99 Eingabe-Tokens einlesen

Sonderzeichen	Escape-Sequenz	Bedeutung
\$	\\$	Beginn und Ende einer Formel
&	\&	Trennzeichen zwischen Tabellenspalten
%	\%	Kommentar
#	\#	Zugriff auf Parameter
_	_	Subscript
{	\{	Anfang eines Arguments
}	\}	Ende eines Arguments
\	<code>\backslash</code>	Teil fast jedes Befehls

Tabelle 1: Sonderzeichen in \LaTeX , deren Escape-Sequenzen und Bedeutung

3. Syntax und Aufbau eines \LaTeX -Dokuments

Ein `.tex`-Dokument besteht aus der “Präambel” und dem Dokumentrumpf. \LaTeX -Befehle werden mit einem Backslash eingeleitet. Der erste Befehl eines Dokuments muß die Angabe der Dokumentklasse sein. Beispiel aus diesem Dokument:

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

Die Angaben in eckigen Klammern stellen optionale Parameter dar, der Text in den geschweiften Klammern ist ein obligatorischer Parameter.

In \LaTeX gibt es Sonderzeichen für Systemfunktionen, die mit einem Backslash (`\`) escaped werden müssen. Um nun diese Zeichen dennoch ausgeben zu können müssen dafür Umschreibungen verwendet werden. In der Tabelle 1 auf Seite 9 sind die Sonderzeichen und deren Escape-Sequenzen angegeben.

3.1. Präambel

Präambel nennt man den Bereich der `.tex`-Datei zwischen dem Beginn der Datei und dem Dokumentrumpf. Hier macht man Einstellungen, die das ganze Dokument betreffen. In der Präambel bindet man auch zusätzliche Pakete ein. Dies ist notwendig, um auf erweiterte Features zugreifen zu können.

3.1.1. Dokumentklassen

Die wichtigsten Dokumentklassen sind:

`article` Gängigste Klasse für technische Artikel. Die höchste Gliederungsebene ist `section`.

Es gibt keine separate Titelseite und keinen Seitenumbruch vor und nach den Inhaltsverzeichnissen.

report Für größere Artikel. Titelseite und Inhaltsverzeichnisse beginnen auf einer separaten Seite. Die höchste Gliederungsebene ist **chapter**

dinbrief Vorlage für einen Geschäftsbrief nach DIN.

book Für Bücher gedacht. Zweiseitiger Druck ist Standard. Es gibt mehr Gliederungsebenen.

slides Dokumentklasse für Overheadfolien. Sie verwendet standardmäßig größere Schriften, die auch aus einiger Entfernung noch sichtbar sein sollen.

Alternativ zu den Standardklassen gibt es die Klassenfamilie “Koma-Script”. Die Klassen (z. B.: `scrartcl`, `scrreprt`, `scrbook`, `scrlettr`) orientieren sich eher an den Normen im deutschsprachigen Raum und es werden Überschriften serifenlos geschrieben.

Dokumentklassen wählt man mit dem Befehl

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

aus. Bis auf die `slides` akzeptieren alle Standardklassen die Optionen `10pt`, `11pt`, `12pt` für die Schriftgröße (10 ist Standard) und `a4paper`, `a5paper`, `b5paper`, `letterpaper`, `legalpaper`, `executivepaper` für die Seitengröße (letter ist Standard).

Verschiedene gebräuchliche Optionen für Dokumentklassen:

- `landscape` Querformat
- `titlepage`, `notitlepage` Soll eine eigene Titelseite erzeugt werden?
- `leqno` Gleichungsnummern auf der linken Seite der Gleichung
- `fleqn` Rechtsbündige Formeln
- `openbib` Das “open” bibliography-Format benutzen.
- `draft`, `final` Übervolle Stellen mit einer Linie markieren/nicht markieren.

Folgende Optionen sind ebenfalls für die Standardklassen (außer `slides`) möglich:

- `oneside`, `twoside` Einseitiges oder Zweiseitiges Layout. Z. B. richtet sich die Ausrichtung der Kopfzeilen und Seitennummern danach. Einseitig ist bis auf `book` Standard.
- `openright`, `openany` Bestimmt, ob neue Kapitel immer auf der rechten Seite begonnen werden. Default ist `openany` bzw. `openright` für `book`.
- `onecolumn`, `twocolumn` Eine oder zwei Spalten pro Seite.

Die Klasse Slides bietet die Option “clock”, um die Zeit am unteren Seitenrand anzuzeigen.

Wenn mehr als eine Option angegeben wird, müssen die Optionen mit einem Beistrich getrennt werden.

3.1.2. Packages

Eine der Stärken von L^AT_EX ist, daß Befehle mit Hilfe von Makros zu neuen Befehlen zusammengestellt werden können, die dann als sogenannte Packages zur Verfügung gestellt werden können. Zusätzliche Pakete werden mit dem Befehl

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

geladen. Wenn man mehr als ein Paket lädt, muß man die Paketnamen entweder mit Beistrichen trennen oder pro Paket einen usepackage-Befehl schreiben.

Hierzulande häufig gebrauchte Packages sind z. B. `graphicx` für das Einbinden von Graphiken, `german` für das Umstellen der automatisch generierten Texte auf deutsch oder `inputenc`, welches es ermöglicht, Umlaute ohne Escape-Sequenzen einzugeben.

Alle Optionen, die man im Befehl `\documentclass` angegeben hat, aber für diese Dokumentklasse unbekannt sind, werden an die geladenen Pakete weitergereicht.

3.1.3. Beispiel

Als Beispiel für eine Präambel jene aus diesem Dokument:

```
\documentclass[a4paper]{scrartcl} % Dokumenttyp bestimmen

%%
%%
%% Zusätzliche Pakete laden und optionale Parameter in
%% eckigen Klammern angeben.
%%
%%
%% Vereinbarung erweiterte Zeichensatztabellen
\usepackage[T1]{fontenc}

%% Skalierbare PostScript Fonts, die in PDF-Viewern besser
%% aussehen
\usepackage{lmodern}

%% Deutsch bei automatisch generierten Texten und Daten
\usepackage[german]{babel}
```

```

% latin1-Zeichentabelle , damit Umlaute direkt eingegeben
% werden koennen
\usepackage[latin1]{inputenc}

% Einbinden von Graphiken ermoeeglichen
\usepackage{graphicx}

% Paket fuer nettere Seitenheader laden
\usepackage{fancyhdr}

% Programmcode einbinden
\usepackage{listings}

% Hyperlinks im Dokument ermoeeglichen
\usepackage[urlcolor=blue, % farbige Hyperlinks
             colorlinks=true, % farbige Querverweise
             linkcolor=blue, % Blau als Farbe fuer Links
             bookmarks, % Navigationsleiste im Acrobat
             bookmarksnumbered=true]{hyperref}

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Standardeinstellungen abaendern
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
\parindent=0cm % Kein Absatzeinzug
\parskip=2mm % Abstand zwischen 2 Absaetzen

\pagestyle{fancy} % Nettere Seitenheader auswaehlen

% Konfiguration des Aussehens der eingebundenen Listings
\lstset{
    language=tex, % Programmiersprache angeben
    basicstyle=\ttfamily, % Schriftschnitt des Codes
    showstringspaces=false, % Spaces nicht markieren
    commentstyle=\sffamily, % Schriftart der Kommentare
    tabsize=3, % Tabs als 3 Leerzeichen
    breaklines=true, % Zu lange Zeilen umbrechen
    frame=trBL % Rahmen um das Listing
}

% Angaben fuer die Titelseite machen
\title{Manuskript zur \LaTeX-üEinführung}
\author{Wolfgang Hackl\\\href{mailto:wolf.hackl@gmx.at}{wolf
.hackl@gmx.at}}

```

Listing 1: Präambel dieses Dokuments. Wie man bei title, sieht kommt das Paket listings nicht mit Umlauten zurecht, welche in Programmcodes auch nicht üblich sind.

3.2. Dokumentrumpf

In den Dokumentrumpf schreibt man das eigentliche Dokument. Zum Rumpf gehört alles, was zwischen den Befehlen

```
\begin{document}  
und  
\end{document}
```

steht.

Der erste Schritt im Dokumentrumpf eines Artikels ist üblicherweise, sich um die Titelseite zu kümmern und einen Platz für die Zusammenfassung (den Abstract) zu reservieren. Danach folgt meistens das Inhaltsverzeichnis, das man von Anfang anlegen lassen kann. Es wird dann bei jedem Generierungslauf aktualisiert. Das nachfolgende Listing zeigt den Anfang eines Dokumentrumpfs. Der letzte Teil ist bereits die erste Überschrift.

```

...
\begin{document}

\maketitle % Titel generieren lassen

% Zusammenfassung des Artikels schreiben
\begin{abstract}
In diesem Dokument werden grundlegende Funktionalitäten von
\LaTeX\space vorgestellt. Es wird erklärt, was man braucht,
um einfache Texte setzen zu können.

Dieser Abstract ist für eine übliche Zusammenfassung etwas zu
kurz geraten. Er hat jedoch eine Daseinsberechtigung, weil man
im Quellcode nachschauen kann, wie das geht.
\end{abstract}

\bigskip % Abstand zwischen Abstract und Logo einfüegen
\begin{center} % Zentrierte Umgebung einschalten
% Grafik lion.png importieren
\includegraphics[keepaspectratio,width=4cm]{lion}
\end{center} % Zentrierte Umgebung ausschalten

\newpage % Seitenumbruch erzwingen
\tableofcontents % Inhaltsverzeichnis generieren lassen
\newpage % Seitenumbruch erzwingen

% Überschrift mit Marke für einen Querverweis
\section{Grundlagen}\label{sec:grundlagen}
...

```

3.3. Gliederungen

Ein Dokument kann mit den sogenannten Gliederungsbefehlen in einzelne Teile, Kapitel, Abschnitte und Unterabschnitte aufgeteilt werden. Das allgemeine Format ist:

```
\befehl[<Verzeichniseintrag>]{<Überschrift>}
```

Normalerweise wird auf den Verzeichniseintrag verzichtet und es erscheint die Überschrift 1:1 sowohl im Inhaltsverzeichnis als auch in der Kopfzeile (sofern man sie mit `\pagestyle{fancy}` auf fancy umgestellt hat). Der Verzeichniseintrag wird nur angegeben, wenn man das Problem hat, daß die Überschrift für die Kopfzeile zu lange ist.

Wenn man z. B. anstatt von `\section{...}` `\section*{...}` schreibt, erhält diese Überschrift keine Nummer und wird nicht im Inhaltsverzeichnis angezeigt.

Die gebräuchlichsten Gliederungsebenen sind:

part Teil (nur book und report)

chapter Kapitel (nur book und report)

section Abschnitt

subsection Unterabschnitt

subsubsection Unter-Unterabschnitt

3.4. Ein Minimaldokument

Ein kleines, gebrauchsfähiges Dokument ist im folgenden Listing enthalten:

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

% Vereinbart erweiterte Zeichensatztabellen
\usepackage [T1] {fontenc}

% Skalierbare PostScript Fonts , die im Acrobat besser
  aussehen
\usepackage {lmodern}

% Deutsch bei automatisch generierten Texten und Daten
\usepackage [german] {babel}

% latin1-Zeichentabelle , damit Umlaute direkt eingegeben
  werden
% koennen
\usepackage [latin1] {inputenc}

% Hyperlinks im Dokument ermoeglichen
\usepackage [colorlinks=true , bookmarks] {hyperref}

% Angaben fuer die Titelseite machen
\title {Minimaldokument}
\author {John Doe}

\begin {document}
\maketitle % Titel generieren lassen
\tableofcontents

\section {Meine erste Section}
```

```

Etwas Text kommt in jedem Dokument vor.

\section{Ein Abschnitt}
Hier steht ein unbedeutender Text, der den Platz etwas üfllt
.

\subsection{Unterabschnitt}
Das hier ist äßgliederungsmig weiter unten

\end{document}

```

Listing 2: Quellcode für ein minimales Dokument.

4. Schriftbild

Globale Änderungen des Aussehens macht man in der Präambel. Gängig sind das Setzen des Absatzeinzuges auf 0 und das Erhöhen des Absatzabstandes. Angloamerikaner machen im Gegensatz zu den Europäern immer einen Tabulatorsprung am Absatzbeginn, dafür ist zwischen den Absätzen kein Abstand.

```

\parindent=0cm % Kein Absatzeinzug
\parskip=2mm % Abstand zwischen 2 Absätzen

```

Wenn man serifenlose Schrift haben möchte, kann man die Standardschriftfamilie auf `sfdefault` setzen. Hiervon sind auch Überschriften etc. betroffen. Durch das Nachladen von `helvet` verwendet man im Dokument die Schriftart Helvetica.

```

% helvetica als Defaultfont
\renewcommand{\familydefault}{\sfdefault}
\usepackage{helvet}

```

4.1. Schriftarten

Das Aussehen von Schriften läßt sich mit folgenden Befehlen beeinflussen:

- `\textbf{fett}` Das Wort ist **fett**
- `\textsl{schräg}` Das Wort ist *schräg*
- `\textsf{serifenlos}` Das Wort ist serifenlos (ohne Schnörkel)

- `\textit{kursiv}` Das Wort ist *kursiv*

Als Hervorhebung kennt man auch `\emph{hervorgehoben}`. `Emph` steht für emphasized (hervorgehoben).

4.2. Schriftgrößen

Die Größe von Schriften läßt sich auf folgende Arten beeinflussen: Man kann eine Umgebung benützen wie `\begin{Large}` und `\end{Large}`, um längere Teile etwas größer oder `tiny` zu schreiben. Die andere Möglichkeit geht für einzelne Worte `{\tiny tiny}`: `tiny`.

5. Querverweise

5.1. Fußnoten

Fußnoten⁶ macht man durch das Einfügen von

```
\footnote[<nummer>]{<Fußnotentext>}
```

in den Text. Üblicherweise läßt man das Argument `<nummer>` weg, wodurch die Fußnoten automatisch durchnummeriert werden, was auch am komfortabelsten ist. Die Fußnote 6 wurde so erzeugt:

```
Fußnoten\footnote{\label{fn:fussnote}Das hier ist eine Fußnote}
macht man durch das Einfügen von...
```

5.2. Referenzen

Will man innerhalb eines Dokuments auf einen bestimmten Abschnitt oder auf eine bestimmte Seite verweisen, so bedient man sich einer Referenz. Man muß eine Marke setzen, um darauf verweisen zu können. Mit

```
\label{<marke>}
```

definiert man so einen Markierungspunkt. Man hat beim Referenzieren die Wahl, mittels

```
\pageref{<marke>}
```

⁶Das hier ist eine Fußnote

die Seitennummer zu erhalten, auf der sich die Markierung “marke” im fertigen Dokument befindet. Alternativ dazu kann man mittels

```
\ref{<marke>}
```

die Abschnitts-, Abbildungs-, Tabellen-, bzw. Formelnummer einfügen. Hierbei hängt die Nummer von der Umgebung, in der das Label verwendet wird, ab. Um beispielsweise auf die Überschrift dieses Abschnitts zu referenzieren, könnte man folgende Befehle absetzen:

```
\subsection{Referenzen}\label{sec:referenzen}
```

```
Das Setzen von Labels und das Benutzen von Querverweisen  
wurde in Abschnitt \ref{sec:referenzen} behandelt.
```

Auf das durchgängige Setzen von Labels bei Überschriften sei hingewiesen. Wenn man die Praxis, nach jeder Überschrift ein gleichnamiges Label zu setzen, durchgängig anwendet, braucht man später nicht lange suchen.

Als hilfreich gilt, daß man für Labels die zu Überschriften gehören, “sec:” benutzt. Labels für Bilder fangen immer mit “fig:” an und die für Tabellen immer mit “tab:”.

6. Nützliche Umgebungen

6.1. Aufzählungen

Eine Aufzählung braucht man in fast jedem technischen Dokument. Wie eine Aufzählung aussieht kann man gleich in Abschnitt 1 sehen. Im Punkt “Präambel” ist sogar eine Unteraufzählung vorhanden. Dies kennzeichnet sich durch die Striche (–) anstatt der Bullets (•) als Aufzählungszeichen. Man setzt Aufzählungen folgendermaßen:

```
\begin{itemize}  
  \item <erster Punkt>  
  ...  
  \item <letzter Punkt>  
\end{itemize}
```

Um statt der Bullets eine durchnummerierte Liste zu erhalten, bedient man sich der Umgebung “enumerate” anstatt “itemize”. Die restliche Syntax ist gleich.

Eine andere Art der Aufzählung nennt sich “description” (siehe Abschnitt 2.3). Hier kann man einen Punkt anführen und ihn anschließend erläutern.

```

\begin{description}
  \item [<erster Punkt>] <ausführliche Beschreibung>
  ...
  \item [<letzter Punkt>] <ausführliche Beschreibung>
\end{description}

```

6.2. Floats

Floats werden auch bewegliche Objekte genannt. Der Sinn dahinter ist, daß Bilder und Tabellen oft sehr groß sind und sie durch \LaTeX optimal positioniert werden. Das bedeutet, daß ein Bild selten exakt dort im Dokument erscheint, wo man es laut Quelltext gesetzt hat. \LaTeX verschiebt es, sodaß der Platz besser ausgenutzt wird, d. h. die Absätze rundherum nicht zerklüftet werden. Damit ein Bild oder eine Tabelle als Float gilt, muß man sie in die Umgebungen “figure” bzw. “table” einbetten:

```

\begin{figure}[<Position>]
  <Bild>
  \caption{<Beschreibung des Bilds>}
  \label{fig:<Marke>}
\end{figure}
\begin{table}[<Position>]
  <Tabelle>
  \caption{<Beschreibung der Tabelle>}
  \label{tab:<Marke>}
\end{table}

```

Die Angabe von <Position> ist ein Vorschlag, das Objekt oben (**t**), unten (**b**), auf einer gesonderten Seite (**p**) oder hier (**h**) zu positionieren. Diese Angaben kann man auch kombinieren, sie werden von \LaTeX aber tatsächlich nur als Vorschlag gewertet.

6.2.1. Bilder

Um eine Grafik einzubinden muß man das Paket `graphicx` in der Präambel laden. Der Befehl zum Einbinden einer Bilddatei lautet:

```
\includegraphics[<optionen>]{datei}
```

Die häufigsten Optionen sind `keepaspectratio`, um das Seitenverhältnis beizubehalten, sowie `width=<Größe>` oder `height=<Größe>`, um die Breite der Grafik im Dokument explizit zu setzen.

Dies alleine genügt, um ein Bild einzubinden. Was noch abgeht sind eine Beschreibung, die Eintragung in das Inhaltsverzeichnis und die Möglichkeit, darauf zu referenzieren. Um das alles zu ermöglichen, setzt man das Bild eben in die Floating-Umgebung “figure”. Ein komplettes Beispiel könnte folgendermaßen aussehen:

```

\begin{figure}[ht]
  \centering
  \includegraphics[height=5cm,keepaspectratio]
    {lion}
  \caption[Löwe]{Der Löwe, der als Maskottchen gilt.}
  \label{fig:loewe}
\end{figure}

```



Abbildung 1: Der Löwe, der als Maskottchen gilt.

Bei diesem Beispiel wird als Positionierungsvorschlag “hier” und “oben” angegeben. Das Bild ist zentriert und 5 cm hoch. Unter dem Bild steht eine ausführliche Beschreibung, während im Inhaltsverzeichnis nur “Löwe” steht. Querverweise referenzieren das Bild anhand der Marke “fig:loewe”. Was dabei herauskommt kann man sich am besten anhand von Bild 1 verinnerlichen.

6.2.2. Tabellen

Tabellen in LaTeX sind etwas umständlich zu realisieren. Man kann zwar alle möglichen Tabellen erzeugen, sollte sich deren Aufbau vorher jedoch gut überlegen.

Das Grundgerüst für eine Tabelle sieht so aus:

```

\begin{tabular}[<Platzierung>]{<Format>}
  <Tabelleninhalt>
\end{tabular}

```

Mit der Angabe der <Platzierung> kann man die Position der Tabelle im Text mit der selben Syntax wie bei den Floats beeinflussen.

Das Format bestimmt das Aussehen der einzelnen Spalten der Tabelle. Mögliche Werte sind in der Tabelle 2 angegeben. Der eigentliche Inhalt wird Zeile für Zeile eingegeben. Zuerst wird der Inhalt der ersten Zelle eingegeben, danach ein Spaltenwechsel mit & erzeugt und dann der Inhalt der zweiten Zelle und so fort. Das Ende der Zeile wird mit

Wert	Wirkung
c	Zentrierter Eintrag
l, r	Links- bzw. rechtsbündiger Eintrag
p{SIZE}	Spalte der Breite SIZE, die einen Absatz enthält. Z.B. p{10cm} wie hier, wo der Inhalt einer Zelle nicht mehr in einer Zeile Platz findet.
	Senkrechter Strich.

Tabelle 2: Formatangaben für das Aussehen von Tabellenspalten.

`\` markiert. Wichtig ist, daß man analog zu HTML auch für leere Zellen ein `&` schreiben muß. Zwischen den Zeilen sind horizontale Linien (`\hrline`) möglich. Das nachfolgende Beispiel für eine Tabelle ist im Gegensatz zu Tabelle 2 kein bewegliches Objekt und steht daher mitten im Text, hat keine Beschreibung und scheint nicht im Tabellenverzeichnis auf:

Lehrstuhl	Professoren
LS2	Wegener
LS8	Morik
LS10	Doberkat

Tabellen können sehr vielseitig gestaltet werden. Durch die komplexe Syntax können sie hier nicht weiter behandelt werden.

6.3. Verbatim

Diese Umgebung benötigt man, um Zeichen “wortwörtlich” darzustellen. Das beste Beispiel dafür sind die \LaTeX -Befehle, die in diesem Dokument erläutert werden, also Sonderzeichen. Text, der in der Umgebung `verbatim` gesetzt wurde, wird im Dokument mit einer Schriftart mit fixer Breite dargestellt. Im fließenden Text schaltet man mit

```
\verb+<zu setzender Text>+
```

in die Verbatim-Umgebung. Die beiden `+` ist der Begrenzer. All

6.4. Mathematische Formeln

Für mathematische Formeln gibt es spezielle Umgebungen. Man hat die Wahl zwischen abgesetzten und im fließenden Text erscheinenden Formeln. Die einzelnen Möglichkeiten sind in Tabelle 3 angegeben.

Wie Formeln im fließenden Text aussehen kann man sich anhand der chemischen Formel für Wasser H_2O verdeutlichen. Wenn die Hypotenuse c ist, dann gilt $c^2 = a^2 + b^2$. Ganz toll sind auch die Formeln 1, 2 und 3.

Syntax	Bedeutung
<code>\begin{math}</code> <code>\end{math}</code>	Formel im fortlaufenden Text
<code>\(... \)</code>	Formel im fortlaufenden Text
<code>\$... \$</code>	Formel im fortlaufenden Text
<code>\begin{displaymath}</code> <code>\end{displaymath}</code>	Abgesetzte Formel ohne Numerierung
<code>\[... \]</code>	Abgesetzte Formel ohne Numerierung
<code>\begin{equation}</code> <code>\end{equation}</code>	Abgesetzte Formel mit Numerierung
<code>\begin{eqnarray}</code> <code>\end{eqnarray}</code>	Abgesetzte Formeln mit Numerierung
<code>\begin{eqnarray*}</code> <code>\end{eqnarray*}</code>	Abgesetzte Formeln ohne Numerierung

Tabelle 3: Syntax für verschiedene Arten von Formeln

$$\left| \sum_{i=1}^n a_i b_i \right| \leq \left(\sum_{i=1}^n a_i^2 \right)^{1/2} \left(\sum_{i=1}^n b_i^2 \right)^{1/2} \quad (1)$$

$$\frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \quad (2)$$

$$\ln \left[\lim_{z \rightarrow 0} \left(1 + \frac{1}{z} \right)^z \right] + (\sin^2 x + \cos^2 x) = \sum_{n=0}^{\infty} \frac{\cosh(y \sqrt{1 - \tanh^2 y})}{2^n} \quad (3)$$

Der Quelltext dazu sieht folgendermaßen aus:

```
\begin{equation}
\left| \sum_{i=1}^n a_i b_i \right|
\le
\left( \sum_{i=1}^n a_i^2 \right)^{1/2}
\left( \sum_{i=1}^n b_i^2 \right)^{1/2}
\label{for:asdf}
\end{equation}
```

```
\begin{equation}
\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}
\label{for:qwer}
\end{equation}
```

```
\begin{equation}
```

```

\ln \left[ \lim_{z \rightarrow 0}
\left(1 + \frac{1}{z}\right)^z \right] +
(\sin^2 x + \cos^2 x) = \sum^{\infty}_{n=0}
\frac{\cosh(y \sqrt{1 - \tanh^2 y})}{2^n}
\label{for:uiop}
\end{equation}

```

Es gibt enorme Mengen an Sonderzeichen für mathematische Formeln. Leider sind Formeln zu kompliziert für eine Einführung. Hier sei auf die weiterführende Literatur verwiesen, z. B. <ftp://server.maths.umanitoba.ca/pub/gratzer/mil/mil.pdf>.

7. Verzeichnisse

Um Verzeichnisse (z. B. das Inhaltsverzeichnis) zu erhalten, genügt ein einfacher Befehl. Das jeweilige Verzeichnis erscheint im Dokument an der Stelle, wo der Befehl angegeben wurde. Folgendes sind die häufigsten Verzeichnisse:

Verzeichnis	Befehl
Inhaltsverzeichnis	<code>\tableofcontents</code>
Abbildungsverzeichnis	<code>\listoffigures</code>
Quellcodeverzeichnis	<code>\lstlistoflistings</code>

Das Inhaltsverzeichnis selbst und die Bibliographie scheinen nicht im Inhaltsverzeichnis auf. Es gibt die Möglichkeit, Einträge in Verzeichnissen zu erzwingen. Das kann man mit den Befehlen

```

\addcontentsline{toc}{chapter}{Inhaltsverzeichnis}
\addcontentsline{toc}{chapter}{Bibliographie}

```

machen. Nützlich ist ein manueller Eintrag im Inhaltsverzeichnis insofern, als man somit durch das gesamte Dokument per Bookmarks navigieren kann. Die Semantik der `addcontentsline`-Befehle ist, daß zur Table of Contents (toc) ein Element der Größenordnung Kapitel (chapter) mit der Bezeichnung Inhaltsverzeichnis hinzugefügt wird.

Am Ende eines Dokument gehört es sich, in einem für das Zielpublikum angemessenen Stil auf verwendete Quellen zu verweisen. Dabei reichen die Spannen von einer formlosen Linksammlung bis zur Angabe in Harvard-Stil oder nach DIN. Für kleinere Dokumente genügt es, die Bibliographie manuell zu schreiben. Wenn man viele und vor allem große Dokumente schreibt, in denen man die selben Quellen verwendet, empfiehlt es sich, eine Literaturdatenbank zu erstellen und die Bibliographie mittels `bibtex` zu verwalten.

A. Linksammlung

Die L^AT_EX-Project-Homepage:

<http://www.latex-project.org/>

The Comprehensive T_EX Archive Network. Bezugsquelle für Software und Dokumentation:

<http://www.ctan.org/>

Homepage der T_EX-Usergroup:

<http://tug.org/>

Homepage der DANTE, Deutschsprachige Anwendervereinigung T_EX e.V.:

<http://www.dante.de/>

Die FAQ der Newsgroup de.comp.text.tex:

<http://www.dante.de/faq/de-tex-faq/html/de-tex-faq.html>

Die FAQ der Newsgroup comp.text.tex:

<http://www.tex.ac.uk/cgi-bin/texfaq2html?introduction=yes>

Befehlszeilen-Referenz:

<http://www.weinelt.de/latex/>

T_EX-Service des Hochschulrechenzentrums der Universität Gießen:

<http://www.uni-giessen.de/hrz/tex/>

L^AT_EX-Workshop Chemnitzer Linux-Tag 2004:

<http://www-user.tu-chemnitz.de/~maku/clt2004-latex/>

Tutorial über Slides in L^AT_EX:

<http://astronomy.sussex.ac.uk/~eddie/soft/tutorial.html>

Tutorial “Making flashy presentations with L^AT_EX”:

<http://www.amath.washington.edu/~medlock/presentation.html>

Homepage des Präsentationspakets “beamer”:

<http://latex-beamer.sourceforge.net>

Dokumentation des Packages pstricks:

<http://www.pstricks.de/>

Homepage des L^AT_EX@TUG-Projekts:

<http://latex.tugraz.at>

Literatur

- [kurzgu] Matthias Kalle Dallheimer (1988): \LaTeX kurz und Gut. O'Reilly, Köln. ISBN 3-89721-204-8
- [de-faq] DE-TeX-FAQ <http://www.dante.de/faq/de-tex-faq/html>
- [index] Der LaTeX-Index - eine Befehlsübersicht im World Wide Web
<http://www.weinelt.de/latex/>
- [vorteil] \LaTeX und MacOS X <http://latex.yauh.de/vorteile.html>
- [lshort] The Not So Short Introduction to $\text{\LaTeX}2_{\epsilon}$ <http://www.ctan.org/tex-archive/info/lshort/english/lshort.pdf>