

SEMINAR MMI - VORLAGE FÜR SCHRIFTLICHE AUSARBEITUNGEN

Max Mustermann

mustermann@mmi.rwth-aachen.de

Abstract

Dieses Dokument dient als Anleitung und als Vorlage für die schriftlichen Ausarbeitungen für Seminare am *Institut für Mensch-Maschine-Interaktion (MMI)*. Wer bisher nicht mit \LaTeX gearbeitet hat, sollte parallel mit der PDF-Version und dem Quellcode dieser Vorlage arbeiten, um so leicht die entsprechenden Befehle für ein gewünschtes Layout zu finden und ein Gespür für den Umgang mit \LaTeX zu entwickeln. Helfen dieses Dokument sowie die angegebene Literatur nicht weiter, stehen natürlich die Betreuer:innen für Fragen bereit.

Keywords: \LaTeX , mmiSeminar-Style, Vorlage

1 Einführung

\LaTeX basiert auf \TeX ¹, einem Textsatzsystem, das von Donald E. Knuth [Knu84] entwickelt und veröffentlicht wurde, um wissenschaftlichen Texten ein einheitliches und gutes Layout zu geben. Aufgrund seiner komplexen Syntax ist die direkte Verwendung von \TeX jedoch nicht leicht, deshalb hat Leslie Lamport [Lam94] das Makro-Paket \LaTeX entwickelt. Dieses definiert Makros, die es dem Benutzer erlauben, komplexere und häufig benötigte Folgen von \TeX -Kommandos zu verwenden. Eine neuere und vereinheitlichte Version ist $\LaTeX 2_{\epsilon}$ das wir hier verwenden.

1.1 Warum \LaTeX

\LaTeX (wegen besserer Lesbarkeit auch einfach LaTeX) ist eine Makro-Sprache und muss ähnlich einer Programmiersprache erlernt werden. Entsprechend muss \TeX -Code erst kompiliert oder „*ghost*“ werden, bevor das tatsächliche Layout sichtbar wird. Die Vorteile der Verwendung von LaTeX für professionelle wissenschaftliche Texte treten besonders bei längeren (etwa 100 Seiten) Texten mit vielen Bildern und Formeln zutage. LaTeX setzt Grundregeln um, an denen sich auch professionelle Layouter orientieren, um einen Text gut lesbar und optisch ansprechend zu gestalten. Zusätzlich wird durch die festen Befehle (z. B. für Kapitelanfänge) ein hohes Maß an Konsistenz erreicht. Arbeiten im technischen Bereich werden daher häufig in \TeX ausgeführt. Zum Beispiel verlangen viele Institute verlangen, dass Studien- und Diplomarbeiten in LaTeX geschrieben sind. Fast alle internationalen Zeitschriften und Konferenzen erwarten \TeX -Dokumente, für die sie entsprechende eigene *Styles* entwickelt haben.

1.2 Vom Quelltext zum PostScript

Dateien rund um \LaTeX

Der Quellcode für jedes LaTeX-Dokument steht in einer *.tex*-Datei. Mit dem Befehl `input` können weitere Dateien eingebunden werden. Dies hat den gleichen Effekt, als ob der Quellcode an Stelle des `input`-Befehls stünde². Befehle in LaTeX beginnen übrigens immer mit einem Backslash `\`.

In wissenschaftlichen Arbeiten werden sehr häufig Ergebnisse anderer Forscher zitiert. Im technischen Bereich geschieht dies durch eine Abkürzung oder Indexnummer, die im *Literaturverzeichnis* wiederzufinden ist. Dort stehen dann genaue Angaben zu der jeweiligen Veröffentlichung. In LaTeX gibt es ein einheitliches Format in dem die Daten einer Veröffentlichung, wie z. B. Autor, Titel, Jahr, usw. gespeichert werden. Mittels verschiedener Bibliography-Styles kann das Layout des Literaturverzeichnisses leicht verändert werden. Die Angaben zu den zitierten Arbeiten stehen in der Bibliography (*.bib*-Datei), die dann von BibTeX ausgewertet wird.

Zur Umsetzung verschiedener Layouts im Text wird ein ähnliches Vorgehen wie für das Literaturverzeichnis angewandt. Zunächst gibt es die *Dokumentenklasse* (Befehl `documentclass` in *frame.tex*), die die grundsätzliche Formatierung festlegt. Dann erweitern die Styles (in *.sty*-Dateien) die Dokumentenklassen und bieten weitere Makros an, die die Arbeit mit LaTeX vereinfachen sollen. Ein Beispiel ist *iso8859-1.sty*: Ohne diesen Style kann man Umlaute nicht direkt in den Text schreiben. Der *mmiSeminar*-Style dagegen dient der einheitlichen Formatierung und

¹spricht: Tech

²siehe auch Datei `frame.tex`

einfachen Erstellung der schriftlichen Ausarbeitungen zur späteren Zusammenführung in einem Seminarband.

Von L^AT_EX zu PostScript

Der erste Schritt um aus dem geschriebenen Quellcode eine formatierte PostScript-Version zu erstellen ist das so genannte „*Texen*“ — das Ausführen des `latex` Befehls. Dies geht entweder über die Konsole mit Angabe des Dateinamens der Master-Datei (hier: `frame`), oder per Menüpunkt im verwendeten Editor. In diesem Schritt werden einige Hilfsdateien mit gleichem Namen wie die Master-Datei erstellt. Der eigentliche Text steht in der `.dvi`-Datei (`dvi` steht für `device independent`). Unter Linux und unter Windows existieren Programme zur Anzeige dieser Dateien.

Je nachdem, welche Änderungen im Quellcode vorgenommen wurden, muss `latex` bis zu dreimal ausgeführt werden. Um alle Verweise zu finden, erstellt das Programm zunächst Tabellen, die im nächsten Schritt verwendet werden. Beim ersten `Texen` wird auch eine Liste der Literaturverweise (`cite`) erstellt. Wurden die `.bib`-Datei verändert muss auch der Befehl `bibtex` Master-Datei ausgeführt werden. Hierauf folgen in der Regel zwei Aufrufe von `latex`, bis alle Referenzen korrekt sind. Komfortable L^AT_EX-Entwicklungsumgebungen unter Linux und unter Windows führen die notwendigen Schritte auf Knopfdruck automatisch aus.

Aus der nun korrekten `.dvi`-Datei wird eine PostScript-Datei (`.ps`) mit dem Befehl `dvips -t a4` Master-File erzeugt. Der Parameter `-t` gibt die Papiergröße A4 an. Die Endung des Master-Files ist wieder wegzulassen. Das Ergebnis kann unter Linux mit `gv` oder unter Windows mit `ghostview` betrachtet werden.

Für das Seminar ist die Befehlsfolge, die bei korrektem Quellcode immer zu einem PostScript führen sollte:

```
latex frame
bibtex frame
latex frame
latex frame
```

1.3 Tools

Die für Linux und Windows zur Verfügung stehenden Hilfsmittel unterscheiden sich, obwohl viele der Linux-üblichen Programme ebenso für Windows erhältlich sind. L^AT_EX kommt aus der Unix-Welt und daher ist hier etwas leichter zu verwenden, insbesondere sind die benötigten Programme bereits installiert. Aber auch unter Windows sind alle Anwendungen, die zum Arbeiten mit L^AT_EX gebraucht werden, kostenlos verfügbar. Die jeweiligen Werkzeuge sind in den nächsten Abschnitten kurz beschrieben.

Unter Linux

Die Programmepakete L^AT_EX, Emacs und `dvips` müssen installiert werden. Emacs bietet für praktisch alle unter Linux üblichen Dateiformate Syntax-Highlighting und besondere Makros, die das Programmieren vereinfachen. Dies trifft auch auf `.tex` und `.bib`-Dateien zu. Die Kombination `Ctrl-C Ctrl-E` erzeugt z. B. beliebige L^AT_EX-Umgebungen. Für BiB_TE_X erscheint ein Menüpunkt, mit dem sich neue Einträge für das Literaturverzeichnis generieren lassen. PostScript-Dateien können mit den kostenlos in der Distribution enthaltenen Programmen `ghostview` und `gv` geöffnet werden. Grafiken können unter Linux mit `xfig` erstellt werden.

Unter Windows

Alle Routinen, die für die Verwendung von LaTeX nötig sind, werden unter Windows im MiKTeX-Paket geliefert (siehe Abschnitt „Links zum Seminar“ auf der Seminar-Seite des MMI). Zur Eingabe des Quellcodes kann prinzipiell jeder Editor verwendet werden, mit dem ASCII-Dateien geschrieben werden können. Empfehlenswert ist aber die Verwendung spezieller Entwicklungsumgebungen wie zum Beispiel TeXstudio (siehe „Links zum Seminar“), die die beschriebenen Prozesse automatisieren und nützliche Funktionen zur Verbesserung des Workflows bieten können. Zum Beispiel ermöglicht es TeXstudio, mit einem Rechtsklick auf die jeweiligen Zeilen im Quellcode direkt zur entsprechenden Stelle in der PDF-Datei zu springen oder umgekehrt, wie in den Abbildungen 1 und 2 zu sehen ist. Die grünen Pfeile in der Werkzeugliste markieren die Schaltflächen für „Build & View“ sowie „Compile“ zum Texen des Quellcodes.

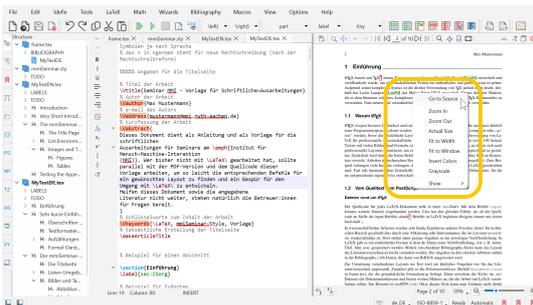


Abbildung 1: Rechtsklick, Go To Source an einer bestimmten Stelle in der PDF-Datei...

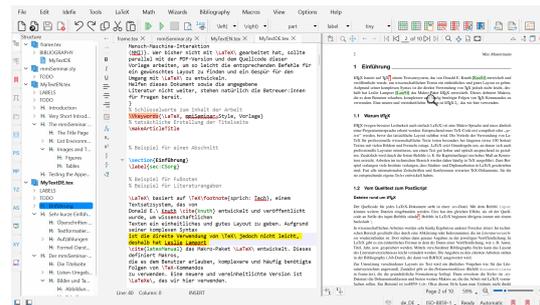


Abbildung 2: ...springt zu den entsprechenden Zeilen im Quellcode und hebt sie hervor.

Viewer wie Yap (zur Anzeige von dvi-Formaten) oder GhostView (zur Anzeige von PostScript-Dateien) müssen im Falle von MiKTeX getrennt installiert werden, stehen dann allerdings in die Entwicklungsumgebungen integriert zur Verfügung.

Grafiken in LaTeX-Dokumenten

Grundsätzlich sollen Grafiken für diese Ausarbeitung als JPG/PNG/GIF eingefügt werden. Es vorteilhaft, die Bilder direkt auf die benötigte Größe zu bringen. Maximale Breite ist die Textbreite von 13,7 cm. Die maximale sinnvolle Auflösung für Bilder ist 150 dpi. Höhere Auflösungen gehen beim Kopiervorgang verloren. Die Bildgröße kann außerdem als Faktor zur Textbreite angegeben werden.

Das Einscannen von Bildern sollte soweit wie möglich vermieden werden. Zum einen ist die Qualität meist nicht gut und die Dateien sehr groß. Zum anderen ist das Verwenden von Teilen aus Büchern selten gestattet. Optimal ist eine Nachahmung der Abbildung mit Verweis auf das Original: nach `\cite{OriginalAutor}`. Sollte dies nicht gehen, kann aus Artikeln das Bild oft mit Grafikprogrammen extrahiert werden. Bei Scans sollte darauf geachtet werden, dass das Rauschen möglichst klein ist, d. h. ein weisser Hintergrund sollte keine schwarzen Pünktchen oder durchscheinende Seiten enthalten.

1.4 Organisation der Dateien

Um diesen Text texen zu können sind eine Reihe von Dateien nötig. Ihre Aufgaben sind im Folgenden kurz beschrieben:

- `frame.tex`
Dient als Ersatz für die Master-Datei des gesamten Seminarbandes. Der Befehl `\input{MyTextDE}` fügt dieses Dokument (`MyTextDE.tex`) ein. „`MyTextDE`“ ist durch den Namen der eigenen Datei zu ersetzen. Weiter unten wird mit dem Befehl `\bibliography{MyBib}` festgelegt, dass die Literaturhinweise in der Datei „`MyBib.bib`“ stehen. Diese sollte bei den Seminararbeiten den gleichen Namen wie der Text haben.
- `MyTextDE.tex`
Der Quelltext dieser Anleitung. Neben dieser Funktion dient die Datei auch als Vorlage für die Seminararbeiten. Mit Hilfe der PostScript Version und des Quellcodes können die Befehle, um ein bestimmtes Layout zu erhalten, leicht gefunden werden. Die eigene Arbeit soll den Namen `nachname.tex` haben (alles klein).
- `MyBib.bib`
Enthält die Literaturangaben zu diesem Text. Die Datei enthält weiterhin Vorlagen für andere Typen von Literaturangaben für die Nicht-Emacs-User. Das eigene Literaturverzeichnis soll den Namen `nachname.bib` haben (alles klein).
- `xxx.tex xxx.sty xxx.bst`
In diesen Dateien werden Styles und Makros definiert, die Erweiterungen oder Veränderungen von LaTeX darstellen. Sie dürfen auf keinen Fall verändert werden.

Weiterhin ist es sehr wichtig, dass alle Bilder als `.jpg`, `.png` oder `.gif` gespeichert sind und sich im Unterverzeichnis `bilder` relativ zu eurem Hauptfile befinden. Wenn also euer `nachname.tex` in `home/username/seminar` steht, gehören die Bilder in `home/username/seminar/bilder`.

2 Sehr kurze Einführung in \LaTeX

Für komplexere Aufgaben sind weitere Nachschlagewerke unverzichtbar. Eine recht umfassende Beschreibung ist kostenlos erhältlich: „*The Not So Short Introduction to $\text{\LaTeX}2\epsilon$* “ [OPHS00], eine Einführung in LaTeX. Für schwierige Formeln ist das \LaTeX -Package sehr zu empfehlen. Das „ *\LaTeX User's Guide*“ [Ame96] beschreibt diese häufig verwendete Erweiterung zu LaTeX. Die beiden Dokumente sind über die Seminar-Seite des MMI verlinkt. Weiterhin gibt es einige durchaus empfehlenswerte Bücher: „ *\LaTeX : A Document Preparation System*“, das Reference Manual [Lam94] und „*The \LaTeX Companion*“ [GMS94], Erklärungen zu verschiedenen Zusatzpackages u. a. \LaTeX . Außerdem gibt es noch die drei Bücher von Kopka zu LaTeX, von denen zunächst nur das erste, „ *\LaTeX Einführung*“ [Kop00], für die einfachen Anwender:innen relevant ist.

2.1 Überschriften und Referenzen

Wie in diesem Beispieldokument zu sehen ist, gibt es für Artikel drei Hierarchieebenen für Überschriften. Sie werden mit den Befehlen `\section{...}`, `\subsection{...}` und `\subsubsection{...}` gesetzt. Dabei ist `...` durch die jeweilige Überschrift zu ersetzen. Innerhalb jedes Blocks kann durch den Befehl `label` eine Marke für spätere Referenzierung

gesetzt werden. In allen solchen Marken sollten wiederum (wie bei Bilddateien) am Anfang stehen. Für diesen Abschnitt ist im Quellcode der Befehl `\label{sec:CSlateg}` zu finden. Mit dem Befehl `\ref{sec:CSlateg}` wird die Abschnittsnummer des angegebenen Abschnitts ausgegeben, in diesem Fall 2. Für jede Überschriftenebene sowie Formeln, Aufzählungspunkte, Definitionen usw. gibt es eigene Zähler, die entsprechend mit `label` gesetzt und mit `ref` wieder abgerufen werden können (siehe Quelltext). Es zählt jeweils die unterste Stufe. Daher ist es sinnvoll, den `label`-Befehl direkt nach einem neuen Block-einleitenden Befehl zu verwenden.

Absätze werden in LaTeX durch eine oder mehrere Leerzeilen erreicht. Alle überflüssigen Leerzeichen und -zeilen werden ignoriert. Befehle wie `\\` (`newline`), `\clearpage` oder `\newpage` sind nur in Sonderfällen zu verwenden und in dieser Arbeit gar nicht.

2.2 Textformatierung

LaTeX bietet einige Möglichkeiten, die Schriftart zu verändern. Diese sollten allerdings sparsam eingesetzt werden, um die Lesbarkeit des Textes nicht zu beeinträchtigen. *Wichtige* oder *neue* Begriffe werden betont (engl. *emphasize*) mit dem Befehl `\emph{...}`. Die Schreibmaschinenschrift `\texttt{...}` wird für Befehle o. ä. verwendet. Bei Befehlen, die LaTeX-Steuerzeichen enthalten empfiehlt sich der Befehl `verb+...+` (vgl. Kommentar im Quellcode in Abschnitt 1.3).

2.3 Aufzählungen

LaTeX stellt im wesentlichen zwei Arten von Aufzählungen zur Verfügung: nummerierte und unnummerierte. Es handelt sich dabei um sogenannte Umgebungen (engl. *environments*), die immer von `\begin{env}` und `\end{env}` umschlossen werden. Dabei ist `env` durch den Namen der jeweiligen Umgebung zu ersetzen. Wer Emacs verwendet, kann mit `Ctrl-C Ctrl-E` leicht *Environments* erstellen. Eine spezifische Aufzählungsumgebung für Algorithmen wird in Abschnitt 3.2 erläutert. Es folgen zwei Beispiele zu den beiden Aufzählungsarten.

Itemize-Umgebung

Dient der nicht nummerierten Aufzählung. Beispiel:

- Item1
- Item2
- Item3

Enumerate-Umgebung

Dient der nummerierten Aufzählung. Beispiel:

1. Item1
2. Item2
3. Item3

- a) Item31
- b) Item32
- c) Item33

Der `label`-Befehl funktioniert auch innerhalb von Aufzählungen (siehe Quellcode): Punkt 3 ist in mehrere Unterpunkte aufgeteilt.

2.4 Formel-Darstellung

LaTeX bietet zahlreiche komfortable Möglichkeiten, um Formeln darzustellen. Da die Darstellung sehr komplexer Formeln ermöglicht wird, existieren entsprechend viele Befehle zu ihrer Beschreibung. Zunächst muss man zwischen mathematischen Ausdrücken im Text, wie $\alpha \in \mathbb{R}$, und abgesetzten Formeln oder Ausdrücken entscheiden. Es folgen einige Beispiele aus anderen Texten, die unter Zuhilfenahme des Quellcodes den Umgang mit Formeln verdeutlichen sollten.

Hier zunächst zwei einfache Formeln:

$$h_i(z) = \frac{1}{\sqrt{2\pi\sigma_i}} \cdot e^{-\left(\frac{z^2}{2\sigma_i^2}\right)} \quad (1)$$

$$o_j = f_{\text{sig}} \left(\sum_{i=1}^N c_{ij} h_i(\|x - w_i\|) \right) \quad (2)$$

Mit `\intertext{...}` lässt sich Text innerhalb ausgerichteter Formelblöcke einfügen.

$$\begin{aligned} w_i(t+1) &= w_i(t) - \alpha(t) [x(t) - w_i(t)] \\ w_j(t+1) &= w_j(t) + \alpha(t) [x(t) - w_j(t)] \end{aligned} \quad (3)$$

where w_i must be of a different class than x and w_j of the same class. If the two closest prototypes are of the same class (training of type (2), the following modification scheme applies (no “window” constraint):

$$w_k(t+1) = w_k(t) + \epsilon\alpha(t) [x(t) - w_k(t)] \quad (4)$$

$$\begin{aligned} \Sigma &= E\{(\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T\} \\ &= \begin{bmatrix} E\{(x_1 - m_1)(x_1 - m_1)\} & \cdots & E\{(x_1 - m_1)(x_n - m_n)\} \\ \vdots & & \vdots \\ E\{(x_n - m_n)(x_1 - m_1)\} & \cdots & E\{(x_n - m_n)(x_n - m_n)\} \end{bmatrix} \\ &= \begin{bmatrix} c_{11} & \cdots & c_{1n} \\ \vdots & & \vdots \\ c_{n1} & \cdots & c_{nn} \end{bmatrix} \end{aligned} \quad (5)$$

Auch Formeln können natürlich Label besitzen, z. B.: Die Modifikation der Gewichte beim LVQ3 Training (4) versteht ohne Erklärung keiner.

3 Der mmiSeminar-Style

Der zunächst von Peter Dörfler für die Seminare am Lehrstuhl für Technische Informatik erstellte Style wird mit dem Wintersemester 06/07 auch für die Seminare des Instituts für Mensch-Maschine-Interaktion eingesetzt. Der Style bietet einige Makros, die das Erstellen von LaTeX Dokumenten vereinfachen. Durch Ändern einiger Verhaltensweisen von LaTeX wird es möglich, den Seminarband als ein einzelnes TeX-Dokument zu behandeln. Weiterhin sind darin einheitliche Formatierungen und das Layout für den Text definiert.

3.1 Die Titelseite

Die Titelseite lässt sich recht schnell erstellen (siehe Quellcode). Es müssen nur die folgenden Werte (eigentlich Makros) gesetzt werden:

- Der Autor des Textes
- Der Titel
- Die e-mail des Autors
- Zusammenfassung
- Schlüsselworte für den Inhalt

Der Befehl `\makeArticleTitle` erstellt dann die Titelseite.

3.2 Listen-Umgebungen im MMI-Style

Aufzählungen

Zwei Arten von Aufzählungen sind hinzugekommen:

1. `narrowenum`
2. `narrowitems`

Die erste der beiden hat direkt hier Anwendung gefunden. Der andere Aufzählungstyp ist in Abschnitt 3.1 zu sehen. Diese Umgebungen sind für Fälle gedacht, in denen die einzelnen Punkte sehr kurz sind. Die normalen Abstände wirken einfach zu groß, wenn pro Punkt nur zwei, drei Worte stehen.

Algorithmus-Umgebung

Die `algorithm`-Umgebung dient dazu Algorithmen darzustellen. Sie funktioniert im Prinzip wie andere Listenumgebungen, hat aber ein etwas anderes Layout. Zusätzlich sind alle weiteren Stufen nach der ersten in die `subalgorithm`-Umgebung zu setzen. Üblicherweise wird ein Algorithmus in eine *floating*-Umgebung gesetzt. Das heisst, dass er nicht unbedingt an der Stelle erscheint, an der der Algorithmus im Text steht, sondern eventuell in den nächsten freien Platz geschoben wird.

Die `figure`-Umgebung (siehe Quellcode) ist für *floating* Bilder, mit der Position als Parameter (**here**, **top**, **bottom**). Der Befehl `caption` legt die Bildunterschrift fest. Diese Umgebung gehört zu Standard-Latex.

```
1. begin begin begin begin begin begin begin begin begin begin
   begin begin begin begin begin begin begin begin begin begin

2.  a) Item21 Item21 Item21 Item21 Item21 Item21 Item21 Item21
     b) Item22 Item22 Item22 Item22 Item22 Item22 Item22 Item22

3. end
```

Abbildung 3: Ein Beispiel-Algorithmus

3.3 Bilder und Tabellen

Sowohl für Tabellen als auch für Abbildungen gibt es im mmiSeminar-Style floating-Umgebungen, die einfacher in der Anwendung sind als die Standardumgebungen.

Abbildungen

Abbildungen sollten im .jpg, .png oder .gif Format vorliegen und in der *figure*-Umgebung per *includegraphics*-Befehl eingebunden werden. Die Bildgröße kann dabei auch relativ zur Textbreite angegeben werden, wie in Bild 4 gezeigt.

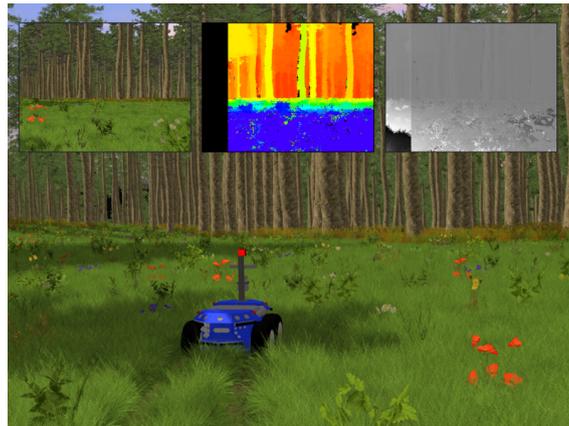


Abbildung 4: Bildunterschrift für das Testbild. Sollte das Bild von einer anderen Quelle stammen, wird hier auch mit cite darauf verwiesen.

Auch in Abbildungen sollte es natürlich möglich sein, allen Text ohne Probleme zu lesen. Leider sieht man oft das Gegenteil.

Tabellen

Die Tabellen-Umgebung *vtable* ist so ähnlich aufgebaut wie die *xxxpsfig*-Befehle weiter oben. Die Parameter sind in Reihenfolge:

1. Position
2. Label

3. Caption
4. eigentliche Tabelle

Für die Labels gilt das gleich wie schon bei den Abbildungen. Der Referenzierungsbefehl ist `tref`. Die eigentliche Tabelle ist am besten an einem Beispiel, Tabelle ??, erklärt. Direkt nach dem Beginn der `tabular`-Umgebung folgt ein Format-String, der die Ausrichtung des Textes in den einzelnen Spalten festlegt. Die Optionen `l`, `r` und `c` stehen für links, rechts und mittig ausgerichtet. Mit `p{Länge}` kann man eine Spalte fester Breite deklarieren. Der senkrechte Strich `|` erzeugt einen ebensolchen. Vom Layout gefälliger und auch lesbarer ist die Verwendung nur nötiger Linien. Die im Beispiel wäre z. B. auch nicht nötig gewesen. Mit `multiline{spalten}{ausrichtung}{text}` kann man einen Text über mehrere Spalten ziehen und ausrichten wie angegeben. Alles weitere steht in der angegebenen Literatur.

A Testing the Appendix ---

Der Anhang wird durch den Befehl `appendixOfArticle` eingeleitet. Auf keinen Fall ist der Befehl `appendix` zu verwenden.

Im Anhang stehen solche Dinge, die nicht in den sonstigen Text passen. Lange Beweise und zusätzliche Beispiele oder Abbildungen gehören dazu. Im Allgemeinen sollte ein Anhang möglichst ganz vermieden werden.

Literaturverzeichnis ---

- [Ame96] American Mathematical Society. *AMS- \TeX Version 1.2 User's Guide*, November 1996.
- [GMS94] Michael Goossens, Frank Mittelbach, and Alexander Samarin. *The \TeX Companion*. Addison-Wesley, Reading, Massachusetts, USA, 1994. ISBN 0-201-54199-8.
- [Knu84] Donald E. Knuth. *The \TeX book*, Volume A of *Computers and Typesetting*. Addison-Wesley, 1984.
- [Kop00] Helmut Kopka. *\TeX Einführung*. Addison-Wesley, Reading, Massachusetts, USA, 3. edition, 2000. ISBN 3-827-31557-3.
- [Lam94] Leslie Lamport. *\TeX : A Document Preparation System*. Addison-Wesley, Reading, Massachusetts, USA, 2. edition, 1994. ISBN 0-201-52984-1.
- [OPHS00] Tobias Oetiker, Hubert Partl, Irene Hyna, and Elisabeth Schlegl. *The Not So Short Introduction to \TeX 2 ϵ , Version 3.16*, September 2000.