
DIPLOMARBEIT

GRAMOC - Gradienten Magnetometer Online Controller

Ausgeführt im Schuljahr 2017/18 von:

Konstruktion des Versuchstandes
Nico Kratky

5CHIF-13

Erstellen der Pumpenkennlinien
Nico LEIDENFROST

5CHIF-14

Betreuer / Betreuerin:

Dr. Michael Stifter

Wiener Neustadt, am 1st July, 2017

Eidesstattliche Erklärung

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche erkenntlich gemacht habe.

Wiener Neustadt, am 1st July, 2017

Verfasser / Verfasserinnen:

Nico Kratky

Nico LEIDENFROST

Contents

Eidesstattliche Erklärung	i
Vorwort	v
Kurzfassung	vi
Abstract	vii
1 Einleitung	1
1.1 Zielsetzung	1
1.2 Warum LaTeX?	1
2 Die Diplomschrift	3
2.1 Elemente der Diplomschrift	3
2.2 Diplomschriften in Englisch	3
3 Zum Arbeiten mit LaTeX	5
3.1 Start	5
3.1.1 Software	5
3.1.2 Literatur	7
3.2 Schrift	8
3.2.1 Schriftarten	8
3.2.2 Texte hervorheben	8
3.3 Textstruktur	9
3.3.1 Absatztrennung	9
3.3.2 Überschriften	10
3.3.3 Listen	10
3.3.4 Absatzformatierung und Zeilenabstand	10
3.3.5 Fußnoten	10
3.3.6 Querverweise	11
3.4 Wortabstand und Punctuation	12
3.4.1 <i>French Spacing</i>	12
3.4.2 Gedanken- und Bindestriche	12
3.4.3 Kommentare	12
3.4.4 Anführungszeichen	13
3.5 Abteilen	13
3.6 Das <code>htl</code> -Paket	14
3.6.1 Einstellungen	14
3.6.2 Definierte Abkürzungen	15
3.6.3 Sprachumschaltung	15

3.6.4	Zusätzliche LaTeX-Pakete	16
3.7	LaTeX-Fehlermeldungen und Warnungen	16
4	Abbildungen, Tabellen, Quellcode	18
4.1	Allgemeines	18
4.2	<i>Let Them Float!</i>	18
4.3	Captions	19
4.4	Abbildungen	19
4.4.1	Wo liegen die Grafikdateien?	20
4.4.2	Grafiken einrahmen	21
4.4.3	Rasterbilder (Pixelgrafiken)	21
4.4.4	Vektorgrafiken	21
4.4.5	TeX-Schriften auch in Grafiken?	22
4.4.6	Abbildungen mit mehreren Elementen	23
4.4.7	Quellenangaben in Captions	23
4.5	Tabellen	23
4.6	Programmtexte	24
4.6.1	Formatierung von Programmcode	24
4.6.2	Platzierung von Programmcode	26
5	Mathem. Formeln etc.	28
5.1	Mathematische Elemente im Fließtext	28
5.2	Freigestellte Ausdrücke	28
5.2.1	Einfache Gleichungen	28
5.2.2	Mehrzeilige Gleichungen	29
5.2.3	Fallunterscheidungen	29
5.2.4	Gleichungen mit Matrizen	30
5.2.5	Verweise auf Gleichungen	30
5.3	Spezielle Symbole	31
5.3.1	Zahlenmengen	31
5.3.2	Operatoren	31
5.3.3	Symbole mit mehreren Zeichen	31
5.3.4	Funktionen	31
5.3.5	Maßeinheiten und Währungen	32
5.3.6	Kommas in Dezimalzahlen (Mathematik-Modus)	32
5.3.7	Mathematik-Werkzeuge	32
5.4	Algorithmen	32
6	Umgang mit Literatur	34
6.1	Allgemeines	34
6.2	Finden von Literatur	34
6.3	Literaturverweise	34
6.3.1	<code>cite</code>	34
6.3.2	Häufige Fehler	35
6.4	Plagiarismus	35
6.5	Literaturverzeichnis	36
6.5.1	Literaturdaten in BibTeX	36
6.5.2	Beispiele	37
6.5.3	Tipps zur Erstellung von BibTeX-Dateien	43

7	Drucken der Diplomarbeit	44
7.1	PDF-Workflow	44
7.2	Drucken	44
7.2.1	Drucker und Papier	44
7.2.2	Druckgröße	44
7.3	Binden	44
7.4	Elektronische Datenträger (CD-R, DVD)	45
8	Hinweise für <i>Word</i>-Benutzer	46
9	Schlussbemerkungen	47
9.1	Lesen und lesen lassen	47
9.2	Checkliste	47
A	Technische Informationen	50
A.1	Aktuelle Dateiversionen	50
A.2	Details zur aktuellen Version	50
A.3	Einstellungen unter Windows	51
A.3.1	TeXnicCenter-Ausgabeprofile	51
B	Inhalt der CD-ROM/DVD	53
B.1	PDF-Dateien	53
B.2	LaTeX-Dateien	53
B.3	Style/Class-Dateien	54
B.4	Sonstiges	54
C	Chronologische Liste der Änderungen	55
D	LaTeX-Quellcode	56
	Bibliography	59

Vorwort

Dies ist **Version 2015/07/09** der LaTeX-Dokumentenvorlage für die Diplomarbeiten an der HTL Wiener Neustadt, basierend auf der Vorlage für Abschlussarbeiten an der FH Hagenberg erstellt von Dr. Wilhelm Burger¹.

Die Verwendung dieser Vorlage ist jedermann freigestellt und an keinerlei Erwähnung gebunden. Allerdings – wer sie als Grundlage seiner eigenen Arbeit verwenden möchte, sollte nicht einfach ("ung'schaut") darauf los werken, sondern zumindest die wichtigsten Teile des Dokuments *lesen* und nach Möglichkeit auch beherzigen. Die Erfahrung zeigt, dass dies die Qualität der Ergebnisse deutlich zu steigern vermag.

Der Quelltext zu diesem Dokument sowie das zugehörige LaTeX-Paket sind in der jeweils aktuellen Version online verfügbar unter

http://unterricht.schermann.org/index.php/Diplomarbeitvorlage_Latex

Trotz großer Mühe enthält dieses Dokument zweifellos Fehler und Unzulänglichkeiten – Kommentare, Verbesserungsvorschläge und passende Ergänzungen sind daher stets willkommen, am einfachsten per E-Mail direkt an mich:

w.schermann@htlwrn.ac.at
Wolfgang Schermann MSc
HTL Wiener Neustadt – Informatik
Austria

Übrigens, hier im Vorwort kann man kurz auf die Entstehung des Dokuments eingehen. Hier ist auch der Platz für allfällige Danksagungen (z. B. an den Betreuer, den Begutachter, die Familie, den Hund, ...), Widmungen und philosophische Anmerkungen. Das sollte man allerdings auch nicht übertreiben und sich auf einen Umfang von maximal zwei Seiten beschränken.

¹<http://www.fh-hagenberg.at/staff/burger/diplomarbeit/>

Kurzfassung

An dieser Stelle steht eine Zusammenfassung der Arbeit, Umfang max. 1 Seite. Im Unterschied zu anderen Kapiteln ist die Kurzfassung (und das Abstract) üblicherweise nicht in Abschnitte und Unterabschnitte gegliedert. Auch Fußnoten sind hier falsch am Platz.

Kurzfassungen werden übrigens häufig – zusammen mit Autor und Titel der Arbeit – in Literaturdatenbanken aufgenommen. Es ist daher darauf zu achten, dass die Information in der Kurzfassung für sich *allein* (d.h. ohne weitere Teile der Arbeit) zusammenhängend und abgeschlossen ist. Insbesondere werden an dieser Stelle (wie u.a. auch im *Titel* der Arbeit und im *Abstract*) normalerweise *keine Literaturverweise* verwendet! Falls man unbedingt solche benötigt – etwa weil die Arbeit eine Weiterentwicklung einer bestimmten, früheren Arbeit darstellt –, dann sind *vollständige* Quellenangaben in der Kurzfassung selbst notwendig, z.B. [ZOBEL J.: *Writing for Computer Science – The Art of Effective Communication*. Springer-Verlag, Singapur, 1997].

Weiters sollte man daran denken, dass bei der Aufnahme in Datenbanken Sonderzeichen oder etwa Aufzählungen mit "Knödelisten" in der Regel verloren gehen. Dasselbe gilt natürlich auch für das *Abstract*.

Inhaltlich sollte die Kurzfassung *keine* Auflistung der einzelnen Kapitel sein (dafür ist das Einleitungskapitel vorgesehen), sondern dem Leser einen kompakten, inhaltlichen Überblick über die gesamte Arbeit verschaffen. Der hier verwendete Aufbau ist daher zwangsläufig anders als der in der Einleitung.

Abstract

This should be a 1-page (maximum) summary of your work in English.

Im englischen Abstract sollte inhaltlich das Gleiche stehen wie in der deutschen Kurzfassung. Versuchen Sie daher, die Kurzfassung präzise umzusetzen, ohne aber dabei Wort für Wort zu übersetzen. Beachten Sie bei der Übersetzung, dass gewisse Redewendungen aus dem Deutschen im Englischen kein Pendant haben oder völlig anders formuliert werden müssen und dass die Satzstellung im Englischen sich (bekanntlich) vom Deutschen stark unterscheidet (mehr dazu in Abschn. ??). Es empfiehlt sich übrigens – auch bei höchstem Vertrauen in die persönlichen Englischkenntnisse – eine kundige Person für das "proof reading" zu engagieren.

Die richtige Übersetzung für "Diplomarbeit" ist übrigens schlicht *thesis*, allenfalls "diploma thesis", auf keinen Fall aber "diploma work" oder gar "dissertation".

Übrigens sollte für diesen Abschnitt die *Spracheinstellung* in LaTeX von Deutsch auf Englisch umgeschaltet werden, um die richtige Form der Silbentrennung zu erhalten, die richtigen Anführungszeichen muss man allerdings selbst setzen (s. dazu Abschnitt ?? bzw. ??).

Chapter 1

Einleitung

1.1 Zielsetzung

Dieses Dokument ist als vorwiegend technische Starthilfe für das Erstellen einer Diplomarbeit mit LaTeX gedacht und ist die Weiterentwicklung einer früheren Vorlage¹ für das Arbeiten mit Microsoft *Word*. Während ursprünglich daran gedacht war, die bestehende Vorlage einfach in LaTeX zu übernehmen, wurde rasch klar, dass allein aufgrund der großen Unterschiede zum Arbeiten mit *Word* ein gänzlich anderer Ansatz notwendig wurde. Dazu kamen zahlreiche Erfahrungen mit Diplomarbeiten in den nachfolgenden Jahren, die zu einigen zusätzlichen Hinweisen Anlass gaben.

Das vorliegende Dokument dient einem zweifachen Zweck: *erstens* als Erläuterung und Anleitung, *zweitens* als direkter Ausgangspunkt für die eigene Arbeit. Angenommen wird, dass der Leser bereits über elementare Kenntnisse im Umgang mit LaTeX verfügt. In diesem Fall sollte – eine einwandfreie Installation der Software vorausgesetzt – der Arbeit nichts mehr im Wege stehen. Auch sonst ist der Start mit LaTeX nicht schwierig, da viele hilfreiche Informationen auf den zugehörigen Webseiten zu finden sind (s. Abschn. ??).

1.2 Warum LaTeX?

Diplomarbeiten, Dissertationen und Bücher im technisch-naturwissenschaftlichen Bereich werden traditionell mithilfe des Textverarbeitungssystems LaTeX [?, ?] gesetzt. Das hat gute Gründe, denn LaTeX ist bzgl. der Qualität des Druckbilds, des Umgangs mit mathematischen Elementen, Literaturverzeichnissen etc. unübertroffen und ist noch dazu frei verfügbar. Wer mit LaTeX bereits vertraut ist, sollte es auch für die Diplomarbeit unbedingt in Betracht ziehen, aber auch für den Anfänger sollte sich die zusätzliche Mühe am Ende durchaus lohnen.

Für den professionellen elektronischen Buchsatz wurde bisher häufig *Adobe Framemaker* verwendet, allerdings ist diese Software teuer und komplex. Eine modernere Alternative dazu ist *Adobe InDesign*, wobei allerdings die Erstellung mathematischer Elemente und die Verwaltung von Literaturverweisen zur Zeit nur rudimentär unterstützt werden.²

Microsoft *Word* gilt im Unterschied zu LaTeX, *Framemaker* und *InDesign* übrigens nicht als professionelle Textverarbeitungssoftware, obwohl es immer häufiger auch von großen Verlagen verwendet wird.³ Das Schriftbild in *Word* lässt – zumindest für das

¹Nicht mehr verfügbar.

²Angeblich werden aber für den (sehr sauberen) Satz in *InDesign* ähnliche Algorithmen wie in LaTeX verwendet.

³Siehe auch <http://latex.tugraz.at/mythen.php>.

geschulte Auge – einiges zu wünschen übrig und das Erstellen von Büchern und ähnlich großen Dokumenten wird nur unzureichend unterstützt. Allerdings ist *Word* sehr verbreitet, flexibel und vielen Benutzern zumindest oberflächlich vertraut, sodass das Erlernen eines speziellen Werkzeugs wie LaTeX ausschließlich für das Erstellen einer Diplomarbeit manchen verständlicherweise zu mühevoll ist. Man sollte es daher niemandem übel nehmen, wenn er/sie sich auch bei der Diplomarbeit auf *Word* verlässt. Im Endeffekt lässt sich mit etwas Sorgfalt (und ein paar Tricks) auch damit ein durchaus akzeptables Ergebnis erzielen. Für alle, die so denken, finden sich in Kap. ?? einige spezielle Hinweise zum Arbeiten mit *Word*. Ansonsten sollten für *Word*-Benutzer aber auch andere Teile dieses Dokuments von Interesse sein, insbesondere die Abschnitte über Abbildungen und Tabellen (Kap. ??) und mathematische Elemente (Kap. ??).

Übrigens, genau hier am Ende des Einleitungskapitels (und nicht etwa in der Kurzfassung) ist der richtige Platz, um die inhaltliche Gliederung der nachfolgenden Arbeit zu beschreiben. Hier soll dargestellt werden, welche Teile (Kapitel) der Arbeit welche Funktion haben und wie sie inhaltlich zusammenhängen. Auch die Inhalte des *Anhangs* – sofern vorgesehen – sollten hier kurz beschrieben werden.

Chapter 2

Die Diplomschrift

2.1 Elemente der Diplomschrift

Jede Diplomschrift ist anders und dennoch sind sich gute Arbeiten in ihrer Struktur meist sehr ähnlich, vor allem bei technischen Themen. Als Startpunkt bewährt hat sich die folgende simple Grundstruktur, die man natürlich variieren und beliebig verfeinern kann:

1. **Einführung und Motivation:** Was ist die Problem- oder Aufgabenstellung und warum sollte man sich dafür interessieren?
2. **Präzisierung des Themas:** Hier beschreibt man den aktuellen Stand der Technik oder Wissenschaft ("State-Of-The-Art"), zeigt bestehende Defizite oder offene Fragen auf und entwickelt daraus die Stoßrichtung der eigenen Arbeit.
3. **Eigener Ansatz:** Das ist natürlich der Kern der Arbeit. Hier wird gezeigt, wie man die vorher beschriebene Aufgabenstellung lässt und – häufig in Form eines Programms¹ – realisiert, ergänzt durch illustrative Beispiele.
4. **Zusammenfassung:** Was wurde erreicht und welche Ziele sind noch offen geblieben, wo könnte man weiter arbeiten?

Natürlich ist auch ein gewisser dramaturgischer Aufbau der Arbeit wichtig, wobei man bedenken muss, dass der Leser in der Regel nur wenig Zeit hat und – anders als etwa bei einem Roman – seine Geduld nicht auf die lange Folter gespannt werden darf. Erklären Sie bereits in der Einführung (und nicht erst im letzten Kapitel), wie Sie an die Sache herangehen, welche Lösungen Sie vorschlagen und wie erfolgreich Sie damit waren.

Übrigens, auch Fehler und Sackgassen darf (und sollte) man beschreiben, ihre Kenntnis hilft oft doppelte Experimente und weitere Fehler zu vermeiden und ist damit sicher nützlicher als jede Schönfärberei.

2.2 Diplomschriften in Englisch

Diese Vorlage ist zunächst darauf abgestellt, dass die Diplomschrift in deutscher Sprache erstellt wird. Vor allem bei Diplomarbeiten, die in Zusammenarbeit mit größeren Firmen oder internationalen Instituten entstehen, ist es häufig erwünscht, dass die Diplomschrift zur besseren Nutzbarkeit in englischer Sprache verfasst wird, und viele Schulen lassen dies in der Regel auch zu.

¹*Prototyp* ist in diesem Zusammenhang ein gerne benutzter Begriff, der im Deutschen allerdings oft unrichtig dekliniert wird. Richtig ist: der *Prototyp*, des *Prototyps*, dem/den *Prototyp* – falsch hingegen z. B.: des *Prototypen*!

Beachten sollte man allerdings, dass das Schreiben dadurch nicht einfacher wird, auch wenn einem Worte und Sätze im Englischen scheinbar leichter "‘aus der Feder'" fließen. Gerade im Bereich der Informatik erscheint durch die Dominanz englischer Fachausdrücke das Schreiben im Deutschen mühsam und das Ausweichen ins Englische daher besonders attraktiv. Das ist jedoch trügerisch, da man die eigene Fertigkeit in der Fremdsprache (trotz der meist langjährigen Schulbildung) häufig überschätzt. Prägnanz und Klarheit gehen leicht verloren und bisweilen ist das Resultat ein peinliches Gefasel ohne Zusammenhang und soliden Inhalt. Sofern man das Englische nicht wirklich gut beherrscht, ist es ratsam, zumindest die wichtigsten Teile der Arbeit zunächst in Deutsch zu verfassen und erst nachträglich zu übersetzen. Besonders vorsichtig sollte man bei der Übersetzung von scheinbar vertrauten Fachausdrücken sein. Zusätzlich ist es immer zu empfehlen, die fertige Arbeit von einem "‘native speaker'" korrigieren zu lassen.

Technisch ist, außer der Spracheinstellung und den unterschiedlichen Anführungszeichen (s. Abschn. ??), für eine englische Arbeit nicht viel zu ändern, allerdings sollte man folgendes beachten:

- Die Titelseite (mit der Bezeichnung "‘Diplomarbeit'") ist für die einzureichenden Exemplare jedenfalls in *deutsch* zu halten, auch wenn der Titel englisch ist.
- Ebenso muss neben dem englischen *Abstract* auch eine deutsche *Kurzfassung* enthalten sein.
- Akademische Titel von Personen haben im Englischen offenbar weniger Bedeutung als im Deutschen und werden daher meist weggelassen.

Chapter 3

Zum Arbeiten mit LaTeX

3.1 Start

LaTeX ist eine in den Naturwissenschaften sehr verbreitete und mittlerweile klassische Textverarbeitungssoftware für das Erstellen großer und komplizierter Dokumente mit professionellem Anspruch. Das Arbeiten mit LaTeX erscheint – zumindest für den ungeübten Benutzer – zunächst schwieriger als mit herkömmlichen Werkzeugen für die Textverarbeitung.

Zum Ersten ist – im Unterschied zu den meisten gängigen Textverarbeitungsprogrammen – LaTeX nicht WYSIWYG¹, sondern es handelt sich um eine *Markup Language* (wie HTML) – noch dazu eine recht komplizierte – und zugehörige Werkzeuge. Ungewohnt erscheinen sicher auch die vermeintlich starken Einschränkungen von LaTeX, insbesondere in Bezug auf die Wahl der Schriften und das Layout. Während man anfangs meint, dass diese Rigidität die eigene Kreativität beschränkt, bemerkt man mit der Zeit, dass es gerade dadurch gelingt, sich stärker auf die Inhalte der Arbeit zu konzentrieren als auf deren äußere Form. Dass am Ende die Form dennoch stimmt, ist allerdings nur dann gewährleistet, wenn man sich bei den eigenen Modifikationen der Formate und Parameter äußerster Zurückhaltung auferlegt, es sei denn, man ist in der Zwischenzeit bereits selbst zum LaTeX-*Guru* avanciert.

Insgesamt lohnt sich der Aufwand, wie viele meinen, zumal die Diplomarbeit in jedem Fall (mit oder ohne LaTeX) ein substantielles Stück Arbeit ist. Es sollte mithilfe von LaTeX ein professionell aussehendes Ergebnis einfacher zu erreichen sein und man wird sich wohl auch einigen Ärger mit Bugs und Unzulänglichkeiten gängiger Software ersparen. Zudem könnte es durchaus sein, dass man nebenbei auch sein Auge für die Feinheiten des Buchsatzes (weiter-) entwickelt.

3.1.1 Software

Zum Arbeiten mit LaTeX benötigt man – neben einem Computer – natürlich Software. Musste man sich früher oft die einzelnen Komponenten von LaTeX mühevoll zusammensuchen und für die eigene Umgebung konfigurieren, gibt es mittlerweile für die wichtigsten Plattformen (Windows, Mac Os, Linux) fertige LaTeX-Installationen, die ohne weiteres Zutun laufen. Die aktuelle Version von LaTeX ist L^AT_EX 2_ε (sprich "LaTeX zwei e"). Zum Arbeiten mit LaTeX benötigt man zwei Dinge:

¹"What You See Is What You Get." Es gibt auch WYSIWYG-Implementierungen für LaTeX, z. B. *Scientific WorkPlace* (<http://www.mackichan.com>) oder *LyX* (<http://www.lyx.org>), die aber teuer bzw. relativ langsam sind.

- LaTeX-Installation (Distribution)
- Texteditor oder Autorenumgebung (Frontend)

Sämtliche Komponenten sind kostenlos und für alle gängigen Plattformen verfügbar.

Windows

Unter *Windows* (XP, Vista, W7) hat sich folgendes Setup bewährt, mit dem u. a. auch dieses Dokument erstellt wurde:

- **LaTeX-Distribution:** *MikTeX 2.8* oder höher. MikTeX enthält auch alle notwendigen Hilfsprogramme, wie beispielsweise *bibtex*.²
- **Frontend:** *TeXnicCenter*.³ Grundsätzlich kann man jeden Text-Editor⁴ verwenden, praktischer ist jedoch eine integrierte LaTeX-Umgebung wie *TeXnicCenter*, die einen auch bei der Dateiverwaltung, der Verarbeitung der Dokumente und der Fehlerbehandlung unterstützt. Für diese Vorlage wird mindestens die Version 2.0 benötigt, da sonst keine Unicode Unterstützung gegeben ist.⁵

Beim erstem Mal sollten *MikTeX* und *TeXnicCenter* in genau dieser Reihenfolge installiert werden.

MikTeX sollte so eingerichtet werden, dass fehlende Pakete automatisch geladen werden. Beim ersten Erstellen werden dann alle notwendigen Pakete aus dem Internet geladen und stehen ab dann lokal zur Verfügung. Das erste Erstellen dauert dadurch allerdings mitunter sehr lange. Sollten trotzdem noch fehlende Pakete wie z. B. *tracklang.sty* angezeigt werden, so muss noch MikTeX aktualisiert werden. Update (Admin) im Startmenü von MikTeX unter Maintenance. Das Programm wählt automatisch die richtigen Pakete zur Aktualisierung aus.

Mac OS

Unter Mac OS wurde die aktuelle Vorlage noch nicht getestet. Der folgende Text bezieht sich auf die alte Vorlage und ist möglicherweise nicht mehr korrekt.

Bewährte und weit verbreitete Frontends für den Mac sind *MacTeX*⁶ und *TeXShop*,⁷ das üblicherweise zusammen mit der TeX-Distribution *TeX Live* verwendet wird. Weit verbreitet ist auch der *TeXworks* Editor, der ebenfalls im MacTeX Package enthalten ist und auch MiKTeX unter Windows beigelegt ist.

Linux

Auch unter **Linux** ist *TeX Live* (s. oben) eine häufig verwendete TeX-Distribution. Als Frontend sind beispielsweise *Lyx*⁸, *Kile*⁹ und *Texmaker*¹⁰ verbreitet. In manchen gängigen Linux-Versionen ist bereits eine komplette LaTeX-Distribution enthalten, sodass im besten Fall überhaupt keine zusätzliche Installation notwendig ist.

²<http://www.miktex.org>

³<http://www.texniccenter.org>

⁴Unter Windows z. B. *UltraEdit* (<http://www.ultraedit.com>) oder *WinEdit* (<http://www.winedit.com>).

⁵<http://www.texniccenter.org>

⁶<http://www.tug.org/mactex> – enthält eine komplette und fertig konfigurierte LaTeX-Installation für Mac OS (ca. 1.15 GB download).

⁷<http://www.uoregon.edu/~koch/texshop/>

⁸<http://www.lyx.org>

⁹<http://kile.sourceforge.net>

¹⁰<http://www.xmlmath.net/texmaker>

Eine interessante Alternative ist die Verwendung von *Eclipse*¹¹ als plattformunabhängiges Frontend (mit dem *TeXlipse*¹² Plugin).

Zum Nachinstallieren zusätzlicher LaTeX-Pakete die nicht in der Standardinstallation enthalten sind gibt es für Linux ebenfalls den MikTeX Package Manager¹³. Allerdings empfiehlt es sich meist die notwendigen Pakete statt dessen über die Softwareverwaltung der Linux-Distribution zu installieren. So kann z.B. in Ubuntu das Paket *texlive-lang-german* installiert werden um die für diese Vorlage notwendige Unterstützung der deutschen Sprache nachzuinstallieren.

Automatische Generierung von LaTeX-Code

Bei der Erstellung von komplizierterem LaTeX-Code kann man sich zum Glück einiger Hilfsmittel bedienen.

- *Online Latex Equation Editor*¹⁴ ist ein einfach zu bedienender Online Editor für mathematische Formeln.
- *Writer2LaTeX*¹⁵ ist ein Export-Filter für OpenOffice-Writer mit dem jede Form von Textdokumenten in LaTeX-Code umgewandelt werden kann.
- *Calc2LaTeX*¹⁶ ist ein Export-Makro für OpenOffice-Calc mit dem Tabellen für das Einfügen in LaTeX erzeugt werden können.
- *Dot2Tex*¹⁷ ist eine Verbindung von Graphviz nach LaTeX. Graphen können in LaTeX eingebunden werden und die Beschriftungen mit LaTeX geschrieben werden.
- *gnuplot*¹⁸ ist ein Programm zur Erstellen von Diagrammen. Über das LaTeX-Package gnuplottex kann es direkt in den LaTeX-Code integriert werden.

3.1.2 Literatur

Es ist müßig, ohne geeignete Literatur mit LaTeX zu beginnen, selbst als fortgeschrittener Benutzer wird man immer wieder auf Hilfe angewiesen sein. Erfreulicherweise ist sehr viel Nützliches auch online verfügbar. Gute Startpunkte sind z. B.

- *L^AT_EX 2_ε-Kurzbeschreibung* von Schmidt et al. [?]
- *The Not So Short Introduction to L^AT_EX 2_ε* von Oetiker et al. [?]

Als mittlerweile bereits klassisches Handbuch zu LaTeX ist

- *A Guide to L^AT_EX 2_ε* von H. Kopka und P. Daly [?]

zu empfehlen, zu dem es für Interessierte auch zwei vertiefende Zusatzbände in Deutsch gibt. Zahlreiche weitere Dokumente zu LaTeX und verwandten Themen finden sich u. a. im Rahmen des *Comprehensive TeX Archive Network* (CTAN) auf

<http://www.ctan.org>¹⁹

<http://dante.ctan.org>²⁰

<http://www.tex.ac.uk>

¹¹<http://www.eclipse.org>

¹²<http://texlipse.sourceforge.net>

¹³http://wiki.ubuntuusers.de/MiKTeX_Package_Manager

¹⁴<http://www.codecogs.com/latex/eqneditor.php>

¹⁵<http://www.ooowiki.de/Writer2LaTeX>

¹⁶<http://www.ooowiki.de/Calc2LaTeX>

¹⁷<http://www.fauskes.net/code/dot2tex>

¹⁸<http://www.gnuplot.info>

¹⁹<http://www.ctan.org/tex-archive/help/Catalogue/bytopic.html>

²⁰<http://dante.ctan.org/tex-archive/help/Catalogue/bytopic.html>

Besonders nützlich sind auch die *Comprehensive List of LaTeX Symbols* [?] und die Beschreibungen wichtiger LaTeX-Pakete, wie

```
babel [?],
graphics, graphicx [?],
fancyhdr [?],
caption [?],
subfigure [?].
```

3.2 Schrift

3.2.1 Schriftarten

LaTeX verwendet normalerweise die Schriften der *Computer Modern* (CM) Serie, die so wie die *TeX*-Software selbst von Donald Knuth²¹ entwickelt wurden. Die drei Basisschrifttypen der CM-Serie in LaTeX sind

Roman	<code>\textrm{Roman}</code>
Sans Serif	<code>\textsf{Sans Serif}</code>
Typewriter	<code>\texttt{Typewriter}</code>

In den Augen vieler Benutzer ist allein die Qualität und Zeitlosigkeit dieser Schriften ein Grund, LaTeX für seriöse Zwecke zu verwenden. Ein weiterer Vorteil der *TeX*-Schriften ist, dass die unterschiedlichen Schriftfamilien und Schnitte bezüglich der Größe sehr gut aufeinander abgestimmt sind, was beim Arbeiten mit verschiedenen *TrueType*-Schriften (z. B. in *Word*) durchaus zum Problem werden kann (s. auch Kap. ??).

Darüber hinaus können aber in LaTeX auch beliebige *PostScript*-Schriften (Type 1) verwendet werden, was allerdings in der Praxis einiges an "Tuning"-Arbeit verlangt. Häufig verwendet werden z. B. *Times* und *Palatino*, derzeit ist aber ein Trend zurück zu den klassischen CM-Schriften zu beobachten.

3.2.2 Texte hervorheben

Texte können auf unterschiedliche Weise aus dem Fließtext hervorgehoben werden.

- Die Auszeichnung in *Kursivschrift* oder "italic" (`\textit{...}`) ist vor allem zum Hervorheben von Betonungen und Zitaten geeignet, aber auch für Produktbezeichnungen, Fremdwörter und Variablen im Text, z. B.

`\textit{Variable}` → *Variable*

- *Slanted* (`\textsl{...}`) bedeutet eine geneigte Schrift und unterscheidet sich damit deutlich von *Italic*. Wird beispielsweise verwendet für die laufenden Kopfzeilen, Produktbezeichnungen und Markennamen – zum Vergleich:

<code>\textrm{Daimler-Chrysler}</code>	→ Daimler-Chrysler
<code>\textsl{Daimler-Chrysler}</code>	→ <i>Daimler-Chrysler</i>
<code>\textit{Daimler-Chrysler}</code>	→ <i>Daimler-Chrysler</i>

- **Boldface** (`\textbf{...}`) wird i. Allg. verwendet für **Überschriften**, Bezeichnungen von **Abbildungen** und **Tabellen**, im Fließtext aber selten:

`\textbf{Überschriften}` → **Überschriften**

²¹<http://www-cs-staff.stanford.edu/~knuth/>

- *Emphasize* (`\emph`) ist normalerweise gleichbedeutend mit `\textit`, wobei `\emph{..}` allerdings auch bei geschachtelten Hervorhebungen und im Bereich anderer Schriftschnitte das "Richtige" tut:

```
\textrm{Du \emph{auch} hier?} → Du auch hier?
\textit{Du \emph{auch} hier?} → Du auch hier?
\textsl{Du \emph{auch} hier?} → Du auch hier?
\textbf{Du \emph{auch} hier?} → Du auch hier?
\texttt{Du \emph{auch} hier?} → Du auch hier?
```

- Unterstreichungen sind ein Relikt aus der Schreibmaschinenära und im modernen Schriftsatz eigentlich überflüssig. Sie sollten daher nur in Ausnahmefällen verwendet werden, z. B.

```
\underline{überflüssig}22
```

3.3 Textstruktur

3.3.1 Absatztrennung

Absätze werden in LaTeX-Quelltext ausschließlich durch das Einfügen einer oder mehrerer **Leerzeilen** voneinander getrennt, es sind also *keinerlei sonstige Steueranweisungen* notwendig!

Besonders die Verwendung von `\\` und `\newline` Anweisungen zur Absatztrennung ist ein häufig zu beobachtender **Fehler**. Vor normalen Absätzen auch *nichts* verloren hat die Anweisung `\paragraph{}` – sie ist in LaTeX (im Unterschied zu HTML) eine Markierung für Überschriften mit Titel (s. unten)!

Üblicherweise wird durch LaTeX zwischen aufeinanderfolgenden Absätzen *kein* zusätzlicher vertikaler Abstand eingefügt.²³ Allerdings wird die *erste* Zeile jedes Absatzes (mit Ausnahme des ersten Absatzes eines Abschnitts) eingerückt, um so die Absatzgrenzen deutlich zu machen. Dieses Schema hat sich nicht nur im traditionellen Buchsatz bewährt²⁴ und sollte auch beibehalten werden, es sei denn man hat wirklich *sehr* gute Gründe dagegen. Für alle übrigen Gliederungen im vertikalen Textfluss sind Überschriften (s. unten) vorgesehen.

Manchmal besteht allerdings der Wunsch, etwa zur Verdeutlichung eines inhaltlichen Sprungs *zwischen* zwei Absätzen einen zusätzlichen Abstand einzufügen, ohne dabei eine neue Überschrift zu setzen. Das kann man gegebenenfalls (wie vor dem aktuellen Absatz passiert) durch

```
\SuperPar Manchmal besteht allerdings der Wunsch, ...
```

erreichen, sollte jedoch sehr sparsam und wirklich **nur in begründbaren Einzelfällen** verwendet werden.²⁵

²²Unterstrichene Texte werden zudem nicht automatisch abgeteilt.

²³Das ist die Standardeinstellung in LaTeX und natürlich abhängig von der verwendeten Dokumentenklasse, Style etc.

²⁴Wer es nicht glaubt, sollte sein Bücherregal (oder notfalls das seiner Eltern) nach Gegenbeispielen durchsuchen.

²⁵Das Makro `\SuperPar` ist in `htl.sty` definiert.

3.3.2 Überschriften

LaTeX bietet – abhängig von der verwendeten Dokumentenklasse – einen Satz vordefinierter Überschriftformate in folgender Ordnung:

```
\part{Titel}26
\chapter{Titel}
\section{Titel}
\subsection{Titel}
\subsubsection{Titel}
\paragraph{Titel}
\subparagraph{Titel}
```

Häufiger Fehler: Bei `\paragraph{}` und `\subparagraph{}` läuft – wie in diesem Absatz zu sehen – der dem Titel folgende Text ohne Umbruch in der selben Zeile weiter, weshalb man im Titel auf eine passende Punctuation (hier z. B. `.`) achten sollte. Der horizontale Abstand nach dem Titel allein würde diesen als Überschrift nicht erkennbar machen.

3.3.3 Listen

Listen sind ein beliebtes Mittel zur Textstrukturierung. In LaTeX sind – ähnlich wie in HTML – drei Arten von formatierten Listen verfügbar: ungeordnete Auflistung ("Knödel-liste"), geordnete Auflistung (Aufzählung) und Beschreibungsliste (Description):

```
\begin{itemize} ... \end{itemize}
\begin{enumerate} ... \end{enumerate}
\begin{description} ... \end{description}
```

Listeneinträge werden jeweils mit `\item` markiert, bei `description`-Listen mit `\item[titel]`. Listen können ineinander verschachtelt werden, wobei sich bei `itemize`- und `enumerate`-Listen die Aufzählungszeichen mit der Schachtelungstiefe ändern (Details dazu in der LaTeX-Dokumentation).

3.3.4 Absatzformatierung und Zeilenabstand

Diplomarbeiten werden – wie Bücher – in der Regel einspaltig und im Blocksatz formatiert, was für den Fließtext wegen der großen Zeilenlänge vorteilhaft ist. Innerhalb von Tabellen kommt es wegen der geringen Spaltenbreite jedoch häufig zu Problemen mit Abteilungen und Blocksatz, weshalb man dort ohne schlechtes Gewissen zum Flattersatz ("ragged right") greifen sollte (wie z. B. in Tab. ?? auf Seite ??).

3.3.5 Fußnoten

Fußnoten können in LaTeX an beinahe jeder beliebigen Stelle, jedenfalls aber in normalen Absätzen, durch die Anweisung

```
\footnote{Fußnotentext}
```

gesetzt werden. Zwischen der `\footnote`-Marke und dem davor liegenden Text sollte grundsätzlich *kein Leerzeichen* entstehen (eventuelle Zeilenumbrüche mit `%` auskommentieren). Die Nummerierung und Platzierung der Fußnoten erfolgt automatisch, sehr große Fußnoten werden notfalls sogar auf zwei aufeinanderfolgende Seiten umgebrochen.

²⁶**part** ist für die Gliederung eines größeren Werks in mehrere Teile vorgesehen und wird üblicherweise bei einer Diplomarbeit (und auch in diesem Dokument) nicht verwendet.

Fußnoten in Überschriften

Auch das braucht man ab und zu, ist aber vor allem deshalb kein so einfacher Fall, weil die Fußnote in einer Überschrift nur an Ort und Stelle aufscheinen darf, nicht aber im *Inhaltsverzeichnis*! Ein konkretes Beispiel dafür ist die Überschrift zu Kapitel ??, die etwa folgendermaßen definiert ist:

```
\chapter[Hinweise für Word-Benutzer]%
{Hinweise für Word-Benutzer%
\protect\footnote{Dieser Abschnitt ....}}%
```

Dabei ist der erste (optionale) Titel [Hinweise für ...] der Eintrag im Inhaltsverzeichnis und im Seitenkopf. Der zweite (identische) Titel {Hinweise für ...} erscheint auf der aktuellen Seite und enthält auch den \footnote{} Eintrag, der allerdings an dieser Stelle durch die Direktive \protect "geschützt" werden muss. Die %-Zeichen sind übrigens notwendig, um eventuelle Leerzeichen, die durch Zeilenumbrüche im Quelltext entstehen, zu eliminieren (diesen Trick braucht man in LaTeX häufig, s. Abschn. ??). Ziemlich kompliziert also, und damit ein weiterer Grund, Fußnoten an solchen Stellen überhaupt zu vermeiden.

Generell sollte man mit Fußnoten sparsam umgehen, da sie den Textfluss unterbrechen und den Leser ablenken. Insbesondere sollten Fußnoten nicht (wie vor allem in manchen sozialwissenschaftlichen Werken gepflegt) derart lang werden, dass sie einen Großteil der Seite einnehmen und damit praktisch ein zweites Dokument bilden.²⁷

3.3.6 Querverweise

Zur Verwaltung von Querverweisen innerhalb eines Dokuments stellt LaTeX einen sehr einfachen Mechanismus zur Verfügung. Zunächst muss jede Stelle (Kapitel, Abschnitt, Abbildung, Tabelle etc.) durch

```
\label{key}
```

markiert werden, wobei *key* ein gültiges LaTeX-Symbol sein muss. Damit Labels (die nur Zahlen sind) nicht verwechselt werden, ist es üblich, sie je nach Bedeutung mit einer unterschiedlichen Prefix zu versehen, z. B.

```
cha:kapitel    ... für Kapitel
sec:abschnitt  ... für Abschnitte (Sections) und Unterabschnitte
fig:abbildung  ... für Abbildungen
tab:tabelle    ... für Tabellen
equ:gleichung  ... für Formeln und Gleichungen
```

Beispiele: \label{cha:Einleitung} oder \label{fig:Screen-1}. Mit den Anweisungen

```
\ref{key}      oder      \pageref{key}
```

kann an beliebiger Stelle im Dokument die zu *key* gehörige Nummer bzw. Seitennummer eingesetzt werden, z. B.

```
.. wie in Kap.~\ref{cha:Einleitung} erwähnt ..
.. der Screenshot auf Seite \pageref{fig:Screen-1} ..
```

²⁷Das führt bei Dokumenten mit vielen Fußnoten bei manchen Lesern angeblich so weit, dass sie aus Neugier (oder Versehen) regelmäßig bei den Fußnoten zu lesen beginnen und dann mühevoll die zugehörigen, kleingedruckten Verweise im Fließtext suchen.

Übrigens werden die Bezeichnungen *Kapitel* und *Abschnitt* auffallend oft falsch verwendet – Kapitel haben ausschließlich "ungebrochene" Nummern:

Richtig: Kapitel 7 oder Abschnitt 2.3.4

Falsch: Kapitel 7.2 oder Abschnitt 5

3.4 Wortabstand und Punctuation

3.4.1 *French Spacing*

Im englischsprachigen Schriftsatz ist es üblich, nach jedem Satzende einen gegenüber dem normalen Wortzwischenraum vergrößerten Abstand einzusetzen. Obwohl dies im Deutschen und Französischen traditionell nicht so ist, wird es wegen der verbesserten Lesbarkeit auch hier manchmal verwendet (nicht in diesem Dokument). Falls man die englische ("nicht-französische") Satztrennung mit zusätzlichem Abstand bevorzugt, ist lediglich die Zeile

```
\nonfrenchspacing
```

am Beginn des Dokuments einzusetzen. In diesem Fall sollte man aber die Interpunktion innerhalb von Sätzen (nach . und :) sorgfältig beachten. Beispielsweise schreibt man "Dr. Mabuse" in der Form

```
Dr.\ Mabuse oder Dr.~Mabuse
```

Im zweiten Beispiel wird mit ~ zudem ein Zeilenumbruch am Leerzeichen verhindert.

3.4.2 Gedanken- und Bindestriche

Die Verwendung der falschen Strichlängen (mit und ohne Zwischenraum) ist ganz allgemein eine häufige Fehlerquelle. Bewusst unterscheiden sollte man zwischen

- kurzen Bindestrichen (wie in "Wagner-Jauregg"),
- Minus-Zeichen, z. B. −7 (erzeugt mit \$-7\$), und
- echten Gedankenstrichen – wie hier (erzeugt mit --).

Für das Setzen von Gedankenstrichen²⁸ gibt es eindeutige Konventionen:

1. Im *Deutschen* setzt man üblicherweise einen von zwei Leerzeichen umgebenen Gedankenstrich²⁹ – wie hier (in LaTeX mit \--). Dieser wird auch für die Angabe von Zahlenintervallen (Seiten 12–19) benutzt.
2. In *englischen* Texten verwendet man einen noch längeren Gedankenstrich³⁰ ohne zusätzliche Leerzeichen—as we should be knowing by now (in LaTeX mit ---).

3.4.3 Kommentare

Textteile können in LaTeX zeilenweise mit % auskommentiert werden. Der einem %-Zeichen nachfolgenden Text wird bis zum nächsten Zeilenende überlesen:

```
Das wird gedruckt. %Dieser Text wird ignoriert.
```

Häufig verwendet werden Kommentarzeichen aber auch zum Ausblenden von *white space*, also Leerzeichen und Zeilenumbrüchen. Folgendes Beispiel zeigt etwa, wie man mit % am Zeilendende das Entstehen eines Leerzeichens vor einer nachfolgenden Fußnotenmarke vermeiden kann:

²⁸Für alle drei gibt es übrigens auch in *Word* entsprechende Sonderzeichen.

²⁹Halbgeviertstrich (*En Dash*).

³⁰Geviertstrich (*Em Dash*).

```
In Österreich isst man sonntags Schnitzel.%
\footnote{Was die allgemein gute Kondition erklärt.}
```

Auf ähnliche Weise kann man das Entstehen von ungewolltem Absatzzwischenraum durch gezielten Einsatz von Kommentarzeilen vermeiden, z.B. vor und nach einem zentrierten Textabschnitt:

```
... normaler Text.
%
\begin{center}
  Dieser Text ist zentriert.
\end{center}
%
Und jetzt geht es normal weiter ...
```

Darüber hinaus bietet die `comment`-Umgebung die Möglichkeit, größere Textblöcke in einem Stück auszublenden:

```
\begin{comment}
Dieser Text ...
... wird ignoriert.
\end{comment}
```

3.4.4 Anführungszeichen

Mit Anführungszeichen geht man aus Gewohnheit meist etwas nachlässig um; auch dabei sind aber die Unterschiede zwischen Deutsch und Englisch zu beachten. Hier die richtige LaTeX-Notation für beide Sprachen:

```
`English' → “English”
"Deutsch" → “Deutsch”
```

Bei richtiger Einstellung werden beispielsweise im TeXnicCenter-Editor und im Eclipse-Plugin Texlipse die entsprechenden Zeichenfolgen automatisch eingesetzt. *Einfache* Anführungszeichen erzeugt man im Englischen analog, im Deutschen benötigt man dafür die Makros `\glq` bzw. `\grq` (German left/right quote):

```
`English' → ‘English’
{\glq}Deutsch{\grq} → ,Deutsch‘
```

3.5 Abteilen

Um ein sauberes Schriftbild zu erreichen sind – speziell im Deutschen wegen der großen Wortlängen – Abteilungen (Silbentrennung, Hyphenation) unerlässlich, entweder manuell durch Einfügen von optionalen Trennzeichen oder automatisch. In LaTeX wird grundsätzlich automatisch abgeteilt, wobei die Sprache am Beginn des Dokuments festgelegt und entsprechende Abteilungsregeln für den gesamten Text verwendet werden.

Besonders bei schmalen Textspalten kann es vorkommen, dass LaTeX keine geeignete Stelle für den Zeilenumbruch findet und den Text über den rechten Rand hinaus laufen lässt. Das ist durchaus beabsichtigt und soll anzeigen, dass an dieser Stelle ein Problem besteht, das man durch manuelles Eingreifen reparieren muss.

Generell sollte man gegenüber der automatischen Abteilung misstrauisch sein und das Endergebnis stets sorgfältig überprüfen. Vor allem Wörter mit Umlauten oder Bindestrichen werden in LaTeX oft unrichtig abgeteilt. Bei Bedarf kann man mit \- gezielt zulässige Abteilungspunkte definieren, wie z. B. in

```
Fach\-hoch\-schul\-kon\-fe\-renz.
```

In echten Problemfällen – etwa bei Schwierigkeiten mit Textelementen, die nicht umgebrochen werden dürfen oder können – kann man LaTeX dazu veranlassen, in einzelnen Absätzen etwas weniger pingelig zu formatieren. Das erreicht man durch

```
\begin{sloppypar}
  Dieser Absatz wird ``schlampig'' (sloppy) gesetzt ...
\end{sloppypar}
```

Der letzte Rettungsanker ist, den betreffenden Absatz so umzuschreiben, dass sich ein passabler Zeilenumbruch ergibt (schließlich ist man ja selbst der Autor und niemandem eine Rechtfertigung schuldig).³¹

3.6 Das htl-Paket

Diesem Dokument angeschlossen sind zwei Dateien, die beide zum Erstellen dieses Dokuments erforderlich sind:

htldipl.cls (class file): definiert die Dokumentenstruktur, Layout und den gesamten Vorspann des Dokuments (Titelseite etc.).

htl.sty (style file): enthält nützliche Definitionen. Diese Datei wird von **htldipl.cls** automatisch geladen, kann aber grundsätzlich auch für andere Dokumente verwendet werden.

3.6.1 Einstellungen

Alle (.tex) Dokumente dieser Klasse beginnen mit der Anweisung

```
\documentclass[language]{htldipl}
```

Mit der Option *language* kann die Hauptsprache des Dokuments spezifiziert werden, die möglichen Werte dafür sind

```
german (default)
english
```

Der vollständige Quelltext für eine entsprechende .tex Hauptdatei ist in Anhang ?? gelistet.

Angaben zur Arbeit

Zum erstellen der Titelseite sind einige Angaben notwendig:

```
\title{Titel der Arbeit}
\author{Autor}
\abteilung{Abteilung}
\studienort{Studienort}
\abgabemonat{Monat der Abgabe}
\abgabejahr{Jahr der Abgabe}
```

³¹ Angeblich waren eigenständige Textänderungen in solchen Fällen auch beim früheren Bleisatz durchaus üblich.

Table 3.1: In `htl.sty` definierte Abkürzungsmakros.

<code>\bzw</code>	bzw.	<code>\ua</code>	u. a.
<code>\bzgl</code>	bzgl.	<code>\Ua</code>	U. a.
<code>\ca</code>	ca.	<code>\uae</code>	u. ä.
<code>\dah</code>	d. h.	<code>\usw</code>	usw.
<code>\Dah</code>	D. h.	<code>\uva</code>	u. v. a.
<code>\ds</code>	d. sind	<code>\uvm</code>	u. v. m.
<code>\evtl</code>	evtl.	<code>\va</code>	vor allem
<code>\ia</code>	i. Allg.	<code>\vgl</code>	vgl.
<code>\sa</code>	s. auch	<code>\zB</code>	z. B.
<code>\so</code>	s. oben	<code>\ZB</code>	Zum Beispiel
<code>\su</code>	s. unten		

Titelseiten

Die ersten Seiten der Arbeit, einschließlich der Titelseite, werden durch die Anweisung

```
\maketitle
```

automatisch generiert, abhängig von den obigen Einstellungen:

<i>Seite</i>	<i>Diplomarbeit</i>
i	Titelseite
ii	Copyright-Seite
iii	Eidesstattliche Erklärung

3.6.2 Definierte Abkürzungen

Es wird im `htl`-Paket weiters eine Reihe von Abkürzungsmakros³² definiert, die das Schreiben vereinfachen und für konsistente Zwischenabstände sorgen (Tab. ??). Bei der Verwendung von Makros ist allgemein zu beachten, dass sie nachfolgende Leerzeichen bisweilen "auf-fressen", sodass vor dem nachfolgenden Text kein Abstand erzeugt wird.³³ Dagegen kann man sich mit `\`-Zeichen oder `{}` behelfen, wie in folgendem (nicht sehr schönen) Beispiel:

```
Mopeds und Autos haben \ia\ zwei {\bzw} vier Räder.
Mopeds und Autos haben i. Allg. zwei bzw. vier Räder.
```

3.6.3 Sprachumschaltung

Für englischsprachige Abschnitte (z. B. das Abstract oder englische Zitate) sollte die *Sprache* von Deutsch auf Englisch umgeschaltet werden, um die richtige Form der Silbentrennung zu erhalten. Damit man nicht versehentlich auf das Rückstellen der Sprache vergisst, sind dafür im `htl`-Paket zwei spezielle *Environments* vorgesehen:

```
\begin{english}
  This is a 1-page (maximum) summary
  of your work in English.
\end{english}
```

³²In Anlehnung an den `jkthesis`-Style von Jochen Küpper (<http://www.jochen-kuepper.de>).

³³Bei den meisten der in `htl.sty` definierten Makros wird dies allerdings durch den Einsatz von `\xspace` verhindert.

Table 3.2: Im `htl`-Paket verwendeten LaTeX-Ergänzungen. Alle sind in gängigen LaTeX Standardinstallationen (z. B. MikTeX) bereits enthalten.

<i>Paket</i>	<i>Funktion</i>
<code>algorithmicx</code>	Beschreibung von Algorithmen
<code>amsmath</code> , <code>amssymb</code>	Mathematische Symbole
<code>amsmath</code>	Mathematischer Satzsatz
<code>babel</code>	Sprachumschaltung
<code>babelbib</code>	Mehrsprachige Literaturverwaltung
<code>caption</code>	Flexiblere Captions
<code>cite</code>	Sortierte Literaturverweise
<code>color</code>	Farbige Textelemente und Hintergrundfarben
<code>marvosym</code>	€-Symbol (<code>\euro</code>)
<code>exscale</code>	Korrekte Schriftgrößen im Math-Modus
<code>fancyhdr</code>	zur Gestaltung Kopfzeilen (header)
<code>float</code>	Verbessertes Float-Handling
<code>fontenc</code>	zur Verwendung der cm-super Type1 Postscript Schriften
<code>graphicx</code>	Einbindung von EPS-Grafiken
<code>hyperref</code>	erzeugt aktive Querverweise im PDF-Dokument
<code>ifthen</code>	für logische Entscheidungen in LaTeX
<code>inputenc</code>	Erweiterter Eingabezeichensatz
<code>listings2</code>	Auflistung von Programmcode
<code>upquote</code>	Gerade Hochkommas in <code>verbatim</code> -Texten
<code>url</code>	Behandlung von URLs im Text
<code>verbatim</code>	Verbesserte <code>verbatim</code> -Umgebung

```
\begin{german}
  Text in Deutsch (wenn die Hauptsprache
  auf Englisch gesetzt ist).
\end{german}
```

Zur Kontrolle lässt sich aktuelle Spracheinstellung übrigens mit dem Makro `\language` anzeigen. An dieser Stelle ergibt das beispielsweise `"english"` (*new german*, d. h. neue deutsche Rechtschreibung).

3.6.4 Zusätzliche LaTeX-Pakete

Für die Verwendung dieses Dokuments ist eine Reihe von zusätzlichen LaTeX-Paketen erforderlich (Tab. ??). Diese Pakete werden am Anfang durch das `htl`-Paket automatisch geladen. Alle verwendeten Pakete sind Teil der LaTeX Standard-Installation, wie z. B. in MikTeX, wo man auch entsprechende Dokumentation findet (meist als DVI-Dateien). Die aktuellen Versionen der Pakete sind online verfügbar, u. a. auf den in Abschn. ?? angegebenen CTAN-Sites.

3.7 LaTeX-Fehlermeldungen und Warnungen

Während des Durchlaufs gibt LaTeX Unmengen von Meldungen aus, die einen in ihrer Fülle zunächst nicht verwirren sollten, z. B.:

```
...
Overfull \hbox (14.43593pt too wide) in paragraph at lines 105--109
\OT1/cm/r/m/n/10.95 F[]ur die Ein-bin-dung von Gra-phi-ken in L[]T[]X wird die V
er-wen-dung des Standard-
[10] [11]
```



```
Overfull \hbox (5.01222pt too wide) in paragraph at lines 148--154
\OT1/cmr/m/n/10.95 wen-di-gen Ras-te-rung kei-nen Sinn, auch bei 1200 dpi-Druck
ern. Spe-zi-ell \OT1/cmr/m/it/10.95 Screen-
...
```

Errors (Fehler) müssen korrigiert werden, wobei einem LaTeX diese Arbeit nicht leicht macht, da manchmal (z. B. wenn eine schließende Klammer } vergessen wurde) das Problem erst viel später im Text lokalisiert wird. In solchen Fällen kann es nützlich sein, das erzeugte Ausgabedokument zu inspizieren um festzustellen, ab welcher Stelle die Ergebnisse aus dem Ruder laufen. Bei kapitalen Fehlern bleibt der LaTeX-Prozessor überhaupt stehen und erzeugt keine Ausgabe (in Verbindung mit einer meist kryptischen Fehlermeldung) – hier hilft meist nur eine genaue Analyse des Quelltexts oder der gerade zuvor durchgeführten Schritte. Ein ausführliches Fehlerprotokoll findet man jeweils in der `.log`-Datei des Hauptdokuments.

Falls keine Fehler mehr angezeigt werden, ist zumindest die syntaktische Struktur des Dokuments in Ordnung. Genauer ansehen sollte man sich die Liste von Meldungen jedoch spätestens beim Abschluss der Arbeit, um übrig gebliebene Probleme, wie überlange Textzeilen, unaufgelöste Verweise und ähnliche zu beseitigen. Am Ende sollte das Ergebnis jedenfalls so aussehen:

```
LaTeX-Result: 0 Error(s), 0 Warning(s), ...
```

Chapter 4

Abbildungen, Tabellen, Quellcode

4.1 Allgemeines

Abbildungen (*figures*) und Tabellen (*tables*) werden üblicherweise zusammen mit einem nummerierten Titel (*caption*) zentriert angeordnet (siehe Abb. ??). Im Text *muss* es zu jeder Abbildung einen Verweis geben und die eigentliche Abbildung sollte erst *nach* dem ersten Verweis platziert werden.

4.2 *Let Them Float!*

Das Platzieren von Abbildungen und Tabellen gehört zu den schwierigsten Aufgaben im Satz, weil diese meist viel Platz benötigen und häufig nicht auf der aktuellen Seite im laufenden Text untergebracht werden können. Diese Elemente müssen daher an eine



C010-0031 ©H. ARMSTRONG ROBERTS All Rights Reserved

Figure 4.1: Newport Beach 1957.

geeignete Stelle auf nachfolgenden Seiten verschoben werden, was manuell sehr mühsam (jedoch in *Word* beispielsweise unerlässlich) ist.

In LaTeX funktioniert das weitgehend automatisch, indem Abbildungen, Tabellen und ähnliche als "Floating Bodies" behandelt werden. Bei der Positionierung dieser Elemente wird versucht, einerseits im Textfluss möglichst wenig Leerraum entstehen zu lassen und andererseits die Abbildungen und Tabellen nicht zu weit von der ursprünglichen Textstelle zu entfernen.

Der Gedanke, dass etwa Abbildungen kaum jemals genau an der gewünschten Stelle und möglicherweise nicht einmal auf derselben Seite Platz finden, ist für viele Anfänger aber offenbar sehr ungewohnt oder sogar beängstigend. Dennoch sollte man zunächst einmal getrost LaTeX diese Arbeit überlassen und *nicht* manuell eingreifen. Erst am Ende, wenn das gesamte Dokument "steht" und man mit der automatischen Platzierung wirklich nicht zurende kommt, sollte man (durch gezielte Platzierungsanweisungen [?, S. 33]) in **Einzelfällen** eingreifen.

4.3 Captions

Bei Abbildungen steht der Titel üblicherweise *unten*, bei Tabellen hingegen – je nach Konvention – *oben* (wie in diesem Dokument) oder ebenfalls *unten*. In LaTeX erfolgt auch die Nummerierung der Abbildungen automatisch, ebenso der Eintrag in das (optionale) Abbildungsverzeichnis¹ am Beginn des Dokuments.

Die Markierung der Captions² erfolgt in LaTeX mithilfe der `\label{}` Anweisung, die unmittelbar auf die `\caption{}` Anweisung folgen muss:

```
1 \begin{figure}
2 \centering
3 \includegraphics[width=.85\textwidth]{CS0031}
4 \caption{Newport Beach 1957.}
5 \label{fig:urlaub}
6 \end{figure}
```

Der Name des Labels (`fig:urlaub`) kann beliebig gewählt werden. Die Kennzeichnung `fig:` ist (wie in Abschn. ?? erwähnt) nur eine nützliche Hilfe, um beim Schreiben verschiedene Arten von Labels besser unterscheiden zu können.

Die Länge der Captions kann dabei sehr unterschiedlich sein. Je nach Anwendung und Stil ergibt sich manchmal eine sehr kurze Caption (Abb. ??) oder eine längere (Abb. ??). Man beachte, wie bei kurzen Captions ein zentrierter Satz und bei langen Captions ein Blocksatz verwendet wird (LaTeX macht das automatisch). Captions sollten *immer* mit einem Punkt abgeschlossen sein.³

4.4 Abbildungen

Für die Einbindung von Grafiken in LaTeX wird die Verwendung des Standard-Pakets `graphicx` [?] empfohlen (wird durch das `htl`-Paket bereits eingebunden). Folgende Bildformate können verwendet werden:

¹Ein eigenes Verzeichnis der Abbildungen am Anfang des Dokuments ist zwar leicht erstellt, in einer Diplomarbeit aber (und eigentlich überall sonst auch) überflüssig. Man sollte es daher weglassen.

²Ausnahmsweise wird das Wort "Caption" im Folgenden ohne deutsche Übersetzung verwendet.

³Kurioserweise verlangen manche Anleitungen genau das Gegenteil, angeblich, weil beim klassischen Bleisatz die abschließenden Punkte im Druck häufig "weggebrochen" sind. Das kann man glauben oder nicht, im Digitaldruck spielt es jedenfalls keine Rolle.



C178-1065 ©H. ARMSTRONG ROBERTS All Rights Reserved

Figure 4.2: Beispiel für einen langen Caption-Text. UNIVAC brachte 1961 mit dem Modell 751 den ersten Hochleistungsrechner mit Halbleiterspeicher auf den Markt. Von diesem Computer wurden in den U.S.A. bereits im ersten Produktionsjahr über fünfzig Exemplare verkauft, vorwiegend an militärische Dienststellen, Versicherungen und Großbanken. Die Ablöse erfolgte zwei Jahre später durch das zusammen mit SPERRY entwickelte Modell 820. Das klingt vielleicht plausibel, ist aber frei erfunden und vermutlich völliger Unsinn.

Rasterbilder:	PNG, JPEG, PDF
Vektorgraphiken:	PDF

4.4.1 Wo liegen die Grafikdateien?

Die Bilder werden üblicherweise in einem Unterverzeichnis (oder in mehreren Unterverzeichnissen) abgelegt, im Fall dieses Dokuments in [images/](#). Dazu dient die folgende Anweisung am Beginn des Hauptdokuments [Diplomarbeit.tex](#) (s. auch Anhang ??):

```
\graphicspath{{images/}}
```

Der (zum Hauptdokument relative) Pfad `graphicspath` kann innerhalb des Dokuments jederzeit geändert werden, was durchaus nützlich ist, wenn man z. B. die Grafiken einzelner Kapitel getrennt in entsprechenden Verzeichnissen ablegen möchte. Die Größe der Abbildung im Druck kann durch Vorgabe einer bestimmten Breite oder Höhe oder eines Skalierungsfaktors gesteuert werden, z. B.:

```
\includegraphics[width=.85\textwidth]{CS1065}
\includegraphics[scale=1.5]{CS1065}
```

Man beachte, dass dabei die Datei-Extension nicht explizit angegeben werden muss. Das ist vor allem dann praktisch, wenn man verschiedene Workflows mit jeweils unterschiedlichen Dateitypen verwendet.

4.4.2 Grafiken einrahmen

Mit dem Makro `\FramePic{}` (definiert in `htl.sty`) kann man optional einen dünnen Rahmen rund um die Grafik erzeugen, z. B.:

```
\FramePic{\includegraphics[height=50mm]{CS1065}}
```

Das wird man üblicherweise nur bei Rasterbildern tun, insbesondere wenn sie zum Rand hin sehr hell sind und ohne Rahmen nicht vom Hintergrund abgrenzbar wären.

4.4.3 Rasterbilder (Pixelgrafiken)

Generell sollte man Bilder bereits vorher so aufbereiten, dass sie später beim Druck möglichst wenig an Qualität verlieren. Es empfiehlt sich daher, die Bildgröße (Auflösung) bereits im Vorhinein (z. B. mit *Photoshop*) richtig einzustellen. Brauchbare Auflösungen bezogen auf die endgültige Bildgröße sind:

- **Farb- und Grauwertbilder:** 150–300 dpi
- **Binärbilder (Schwarz/Weiß):** 300–600 dpi

Eine wesentlich höhere Auflösung macht aufgrund der beim Laserdruck notwendigen Rasterung keinen Sinn, auch bei 1200 dpi-Druckern. Speziell *Screenshots* sollte man nicht zu klein darstellen, da sie sonst schlecht lesbar sind (max. 200 dpi, besser 150 dpi). Dabei ist zu bedenken, dass die Arbeit auch als Kopie in allen Details noch gut lesbar sein sollte.

JPEG-Problematik

Keinesfalls sollte man Bilder, die für den Einsatz in Druckdokumenten gedacht sind, mit verlustbehafteten Kompressionsverfahren abspeichern. Insbesondere sollte man die Verwendung von JPEG möglichst vermeiden, auch wenn viele Dateien dadurch wesentlich kleiner würden. Eine Ausnahme ist, wenn die Originaldaten nur in JPEG vorliegen und für die Einbindung nicht bearbeitet oder verkleinert wurden. Ansonsten sollte man immer PNG verwenden.

Besonders gerne werden farbige **Screenshots** einer JPEG-Kompression⁴ unterzogen, obwohl deren verheerende Folgen auch für jeden Laien sichtbar sein sollten (Abb. ??).

4.4.4 Vektorgrafiken

Für schematische Abbildungen (z. B. Flussdiagramme, Entity-Relationship-Diagramme oder sonstige strukturelle Darstellungen) sollte man unbedingt Vektorgrafiken verwenden. Gerasterte Grafiken, wie man sie üblicherweise als GIF- oder PNG-Dateien auf Webseiten findet, haben in einem Druckdokument nichts zu suchen, notfalls muss man sie mit einem entsprechenden Werkzeug *neu* zeichnen (natürlich unter Angabe der ursprünglichen Quelle).

In diesem Fall kommt als Datenformat nur PDF in Frage, dieses bietet sich aber auch in anderen Umgebungen als universelles Vektor-Format an. Zur Erstellung von PDF-Vektorgrafiken benötigt man ein geeignetes Grafikprogramm, z. B. *Freehand* von *Macromedia* oder *Illustrator* von *Adobe*. Manche gängige Grafikprogramme unterstützen allerdings keinen direkten Export von PDF-Dateien oder erzeugen unbrauchbare Dateien. Vor der Entscheidung für eine bestimmte Zeichensoftware sollte man das im Zweifelsfall ausprobieren. PDF kann im Notfall über einen entsprechenden Druckertreiber erzeugt werden.

⁴Das JPEG-Verfahren ist für natürliche Fotos konzipiert und dafür auch gut geeignet, seine undifferenzierte Verwendung ist aber zu einer globalen Plage geworden.

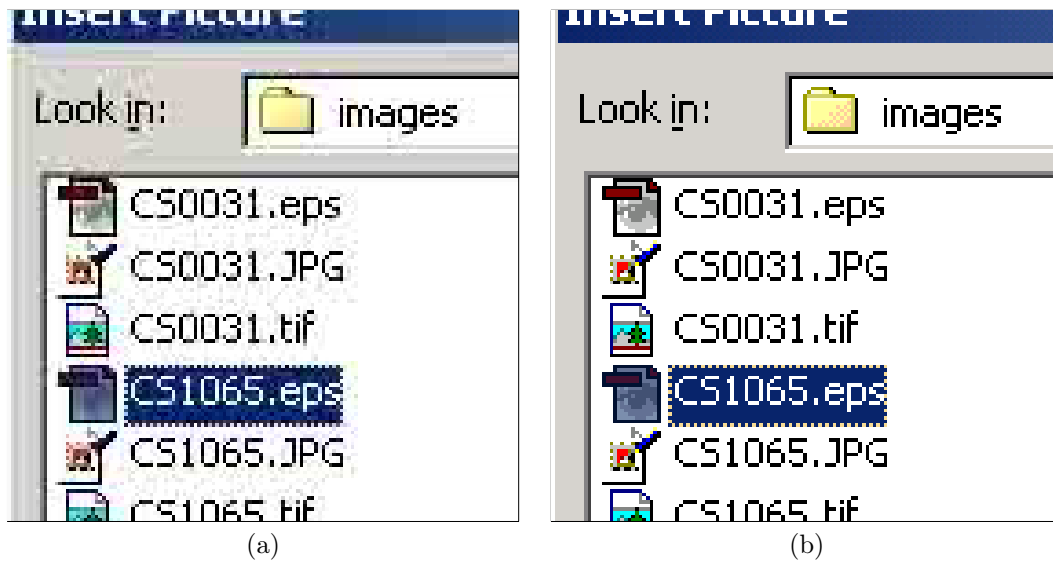


Figure 4.3: Typischer JPEG-Pfusch. Screenshots und ähnliche im Original verfügbare Rasterbilder sollten für Druckdokumente *keinesfalls* mit JPEG komprimiert werden. Das Ergebnis (a) sieht gegenüber dem unkomprimierten Original (b) nicht nur schmutzig aus, sondern wird auch schnell unleserlich.

Einbettung von Schriften

Die Wiedergabe von Textelementen ist abhängig von der auf dem Computer (oder Drucker) installierten Schriften und der Form der Schrifteinbettung im Quelldokument. Die korrekte Darstellung am Bildschirm eines Computers bedeutet nicht, dass dasselbe Dokument auf einem anderen Computer oder Drucker genau so dargestellt wird. Dieser Umstand ist besonders wichtig, wenn Druckdokumente online zur Verfügung gestellt werden. Kontrollieren Sie daher genau, ob die innerhalb Ihrer Grafiken verwendeten Schriften auch exakt wie beabsichtigt im Ausdruck aufscheinen.

Strichstärken – *Hairlines* vermeiden!

In Grafik-Programmen wie *Freehand* und *Illustrator*, die sich im Wesentlichen an der *PostScript*-Funktionalität orientieren, ist es möglich, Linien bzgl. ihrer Stärke als "Hairline" zu definieren. Im zugehörigen *PostScript*-Code wird dies als `linewidth` mit dem Wert 0 ausgedrückt und sollte am Ausgabegerät "möglichst dünne" Linien ergeben. Das Ergebnis ist ausschließlich vom jeweiligen Drucker abhängig und somit kaum vorhersagbar. **Fazit:** Hairlines vermeiden und stattdessen immer konkrete Strichstärken ($\geq 0.25\text{pt}$) einstellen!

4.4.5 TeX-Schriften auch in Grafiken?

Während man sich bei Abbildungen, die mit externen Grafik-Programmen erzeugt werden, meist mit ähnlich aussehenden Schriften (wie *Times-Roman* oder *Garamond*) abhilft, besteht bei Puristen oft der verständliche Wunsch, die *Computer-Modern* (CM) Schriftfamilie von TeX/LaTeX auch innerhalb von eingebetteten Grafiken einzusetzen.

Table 4.1: Prozessor-Familien im Überblick.

	<i>PowerPC</i>	<i>Pentium</i>	<i>Athlon</i>
Manufacturer	Motorola	Intel	AMD
Speed	high	medium	high
Price	high	high	medium

BaKoMa-Schriften

Glücklicherweise stehen einige Portierungen von CM als *TrueType*-Schriften zur Verfügung, die man auch in herkömmlichen DTP-Anwendungen unter *Windows* und *Mac OS* verwenden kann. Empfehlenswert ist vor allem die *BaKoMa Fonts Collection*⁵, die neben den CM-Standardschriften auch die mathematischen Schriften der AMS-Familie enthält und zudem kostenfrei ist. Natürlich müssen die TrueType Schriften vor der Verwendung zunächst auf dem eigenen PC installiert werden.

4.4.6 Abbildungen mit mehreren Elementen

Werden mehrere Bilder oder Grafiken zu einer Abbildung zusammengefasst, verwendet man üblicherweise eine gemeinsame Caption, wie in Abb. ?? dargestellt. Im Text könnte ein Verweis auf einen einzelnen Teil der Abbildung, etwa das einreihige Rollenlager in Abb. ?? (c), so aussehen:

Abb. ~\ref{fig:Bearings}\thinspace(c)

Für kompliziertere Abbildungen sollte man die Verwendung des `subfigure`-Pakets [?] in Betracht ziehen.

4.4.7 Quellenangaben in Captions

Wenn Bilder, Grafiken oder Tabellen aus anderen Quellen verwendet werden, dann muss ihre Herkunft in jedem Fall klar ersichtlich gemacht werden, und zwar am besten direkt in der Caption. Verwendet man beispielsweise eine Grafik aus einem Buch oder einer sonstigen zitierfähigen Publikation, dann sollte man diese in das Literaturverzeichnis aufnehmen und wie üblich mit `\cite{..}` zitieren, wie in Abb. ?? demonstriert. Weitere Details zu dieser Art von Quellenangaben finden sich in Kap. ??.

Bei Bildern aus dem Internet ist es hingegen ratsam, die zugehörige Website *nicht* in das Literaturverzeichnis aufnehmen, sondern (mit `\url{..}`) direkt in der Caption oder in einer Fußnote anzugeben (siehe Abschn. ??).

4.5 Tabellen

Tabellen werden häufig eingesetzt um numerische Zusammenhänge, Testergebnisse etc. in übersichtlicher Form darzustellen. Ein einfaches Beispiel ist Tab. ??, der LaTeX-Quelltext dazu findet sich in Prog. ??.

Manchmal ist es notwendig, in Tabellen relativ viel Text in engen Spalten unter zu bringen, wie in Tab. ?. In diesem Fall ist es sinnvoll, auf den Blocksatz zu verzichten und

⁵Von Basil K. Malyshev – die BaKoMa-Fonts liegen dieser Vorlage bei, ansonsten findet man sie z. B. unter www.ctan.org/tex-archive/fonts/cm/ps-type1/bakoma/.

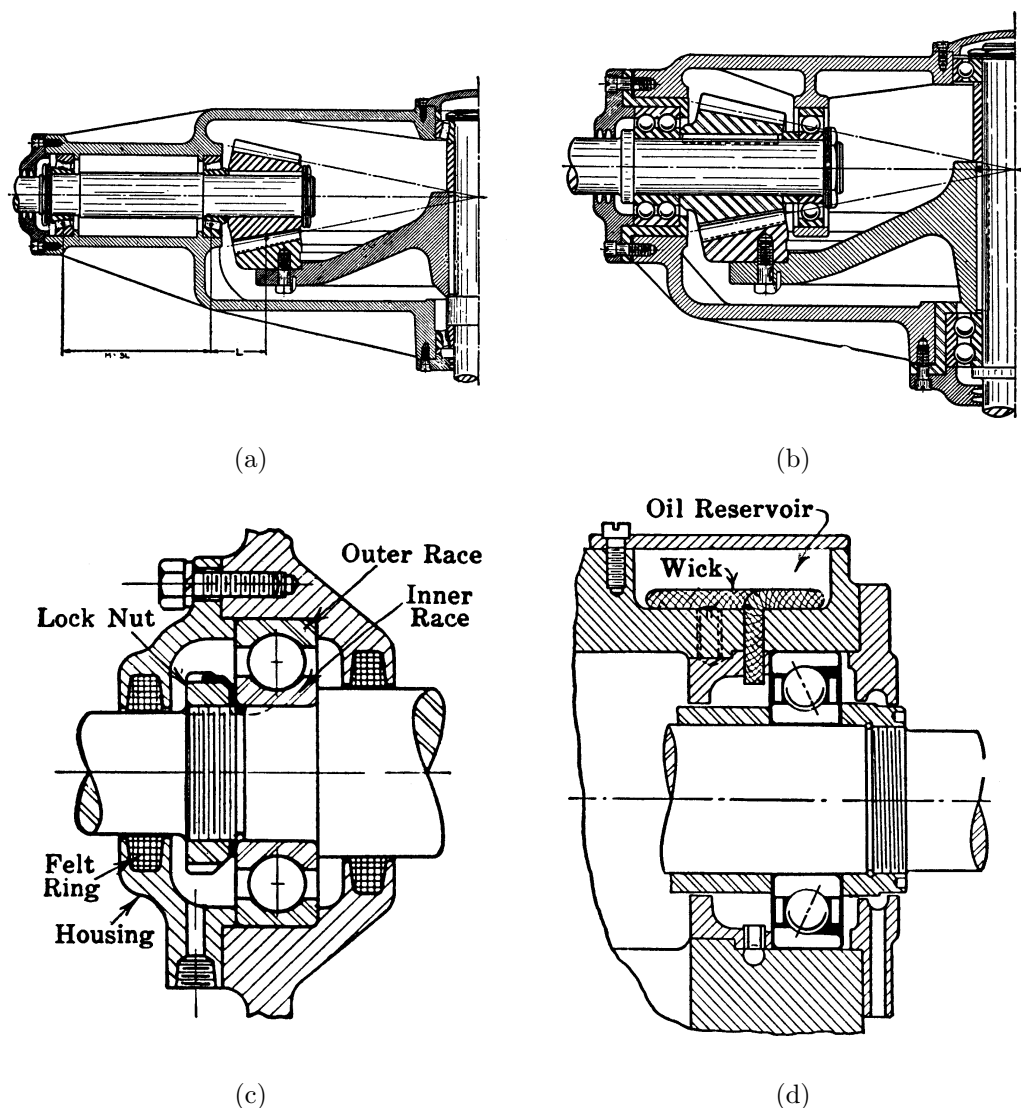


Figure 4.4: Diverse Maschinenelemente. Overhang Mounting (a), Straddle Mounting (b), einreihiges Rollenlager (c), Schmierung von Rollenlagern (d). Aus [?].

gleichzeitig die strengen Abteilungsregeln zu lockern. Details dazu finden sich im zugehörigen LaTeX-Quelltext.

4.6 Programmtexte

Die Einbindung von Programmtexten (source code) ist eine häufige Notwendigkeit, vor allem natürlich bei Arbeiten im Bereich der Informatik.

4.6.1 Formatierung von Programmcode

Es gibt für LaTeX spezielle Pakete zur Darstellung von Programmen, die u. a. auch die automatische Nummerierung der Zeilen vornehmen. Zu empfehlen ist hier insbesondere das `listings2`-Package⁶. In `htl.sty` sind damit folgende Umgebungen definiert:

⁶Aktuell noch als Beta-Version, ist notwendig zur Unterstützung von UTF-8.

Program 4.1: LaTeX Quelltext zu Tab. ???. Die Erzeugung des dargestellten Listings selbst ist in Abschn. ?? beschrieben.

```

1 \begin{table}
2 \caption{Prozessor-Familien im "Überblick.}
3 \label{tab:processors}
4 \centering
5 \setlength{\tabcolsep}{5mm} % separator between columns
6 \def\arraystretch{1.25} % vertical stretch factor
7 \begin{tabular}{|r||c|c|c|} \hline
8 & \emph{PowerPC} & \emph{Pentium} & \emph{Athlon} \\ \hline
9 \hline
10 Manufacturer & Motorola & Intel & AMD \\ \hline
11 \hline
12 Speed & high & medium & high \\ \hline
13 \hline
14 Price & high & high & medium \\ \hline
15 \hline
16 \end{tabular}
17 \end{table}

```

Table 4.2: Beispiel für eine Tabelle mit mehrzeiligem Text in engen Spalten. Hier werden die Zeilen für den Blocksatz zu kurz, daher wird linksbündig gesetzt (im "Flattersatz").

<i>Method</i>	<i>Implem.</i>	<i>Features</i>	<i>Status</i>
polygon shading	SW/HW	flat-shaded polygons	
flat shading with z-buffer	SW/HW	depth values	
goraud shading with z-buffer	SW/HW	smooth shading, simple fog, point light sources	SGI entry models
phong shading with z-buffer	SW/HW	highlights	
texture mapping with z-buffer	SW/HW	surface textures, simple shadows	SGI high end, flight simulators

```

C (ANSI):    \begin{CCode}      ... \end{CCode}
C++ (ISO):   \begin{CppCode}    ... \end{CppCode}
Java:        \begin{JavaCode}   ... \end{JavaCode}
LaTeX:       \begin{LaTeXCode}  ... \end{LaTeXCode}
PHP:         \begin{PhpCode}     ... \end{PhpCode}
Python:      \begin{PythonCode} ... \end{PythonCode}
C#:          \begin{CSharpCode}  ... \end{CSharpCode}
Generisch:   \begin{GenericCode} ... \end{GenericCode}

```

Der Quellcode innerhalb dieser Umgebungen wird in der jeweiligen Programmiersprache interpretiert, wobei Kommentare erhalten bleiben. Diese Umgebungen können sowohl alleinstehend (im Fließtext) oder innerhalb von Float-Umgebungen (insbes. `program`) verwendet werden. Im ersten Fall wird der Quelltext auch über Seitengrenzen umgebrochen. Mit `/+ ... +/` ist eine Escape-Möglichkeit nach LaTeX vorgesehen, die vor allem zum Setzen von Labels für Verweise auf einzelne Programmzeilen nützlich ist, z. B.

```
/+ \label{ExampleCodeLabel} +/
```

Program 4.2: Beispiel für die Auflistung von Programmcode als Float-Element.

```

1 import ij.ImagePlus;
2 import ij.plugin.filter.PlugInFilter;
3 import ij.process.ImageProcessor;
4
5 public class My_Inverter implements PlugInFilter {
6
7     String title = ""; // just to test printing of double quotes
8
9     public int setup (String arg, ImagePlus im) {
10         return DOES_8G; // this plugin accepts 8-bit grayscale images
11     }
12
13     public void run (ImageProcessor ip) {
14         int w = ip.getWidth();
15         int h = ip.getHeight();
16
17         /* iterate over all image coordinates */
18         for (int u = 0; u < w; u++) {
19             for (int v = 0; v < h; v++) {
20                 int p = ip.getPixel(u, v);
21                 ip.putPixel(u, v, 255-p); // invert:  $I'(u,v) \leftarrow 255 - I(u,v)$ 
22             }
23         }
24     }
25
26 } // end of class My_Inverter

```

Ein Beispiel mit Java ist in Prog. ?? gezeigt, wobei der oben angeführte Label in Zeile ?? steht. Man beachte, dass innerhalb der Kommentare auch mathematischer Text (wie etwa in Zeile ?? von Prog. ??) stehen kann.

4.6.2 Platzierung von Programmcode

Da Quelltexte sehr umfangreich werden können, ist diese Aufgabe nicht immer leicht zu lösen. Abhängig vom Umfang und vom Bezug zum Haupttext gibt es grundsätzlich drei Möglichkeiten zur Einbindung von Programmtext:

- a) im laufenden Text für kurze Programmstücke,
- b) als Float-Element (**program**) für mittlere Programmtexte bis max. eine Seite oder
- c) im Anhang (für lange Programmtexte).

Programmtext im laufenden Text

Kurze Kodesequenzen kann man ohne weiteres im laufenden Text einbetten, sofern sie an den gegebenen Stellen von unmittelbarer Bedeutung sind. Die folgende (rudimentäre) Java-Methode `extractEmail` sucht nach einer E-Mail Adresse in der Zeichenkette `line`:

```

1 static String extractEmail(String line) {
2     line = line.trim(); // find the first blank
3     int i = line.indexOf(' ');
4     if (i > 0)
5         return line.substring(i).trim();
6     else

```

```

7         return null;
8     }

```

Dieses Kodestück wurde mit

```

\begin{JavaCode}
static String extractEmail(String line) {
    line = line.trim(); // find the first blank
    ...
}
\end{JavaCode}

```

erstellt (siehe Abschn. ??). In-line Programmstücke sollten maximal einige Zeilen lang sein und nach Möglichkeit nicht durch Seitenumbrüche geteilt werden. Um auch längere Programmzeilen unterzubringen, empfiehlt es sich, dafür eine entsprechend kleine Schriftgröße zu wählen (als Standardgröße ist `footnotesize` eingestellt).

Programmtexte als Float-Elemente

Sind längere Kodesequenzen notwendig, die in unmittelbarer Nähe des laufenden Texts stehen müssen, sollte man diese genauso wie andere Abbildungen als Float-Elemente behandeln. Diese Programmtexte sollten den Umfang von einer Seite nicht übersteigen. Im Notfall kann man auch bis zu zwei Seiten in aufeinanderfolgende Abbildungen packen, jeweils mit eigener Caption. In `htl.sty` ist eine neue Float-Umgebung `program` definiert, die analog zu `table` verwendet wird:

```

\begin{program}
\caption{Der Titel zu diesem Programmstück.}
\label{prog:xyz}
\begin{JavaCode}
    class IrgendWas {
        ...
    }
\end{JavaCode}
\end{program}

```

Wenn man möchte, kann man in diesem Fall die Caption auch unten anbringen (aber konsistent und nicht gemischt). Natürlich darf man auch hier nicht mit einer linearen Abfolge im fertigen Druckbild rechnen, daher sind Wendungen wie "... im folgenden Programmstück ..." zu vermeiden und entsprechende Verweise einzusetzen. Beispiele dafür sind die Programme ?? und ??.

Programmtext im Anhang

Für längere Programmtexte, speziell wenn sie vollständige Implementierungen umfassen und im aktuellen Kontext nicht unmittelbar relevant sind, muss man zur Ablage in einem getrennten Anhang am Ende des Dokuments greifen. Für Hinweise auf bestimmte Details kann man entweder kurze Ausschnitte in den laufenden Text stellen oder mit entsprechenden Seitenverweisen arbeiten. Ein solches Beispiel ist der LaTeX-Quellcode in Anhang ?? (Seite ??).⁷

⁷Grundsätzlich ist zu überlegen, ob die gedruckte Einbindung der gesamten Programmtexte einer Implementierung für den Leser überhaupt sinnvoll ist, oder ob man diese nicht besser elektronisch (auf CD-ROM) beifügt und nur exemplarisch beschreibt.

Chapter 5

Mathematische Formeln, Gleichungen und Algorithmen

Das Formatieren von mathematischen Elementen gehört sicher zu den Stärken von LaTeX. Man unterscheidet zwischen mathematischen Elementen im Fließtext und freistehenden Gleichungen, die in der Regel fortlaufend nummeriert werden. Analog zu Abbildungen und Tabellen sind dadurch Querverweise zu Gleichungen leicht zu realisieren. Hier nur einige Beispiele und spezielle Themen, vieles weitere dazu findet sich z. B. in [?, Kap. 5] oder [?].

5.1 Mathematische Elemente im Fließtext

Mathematische Symbole, Ausdrücke, Gleichungen etc. werden im Fließtext durch paarweise `$... $` markiert. Hier ein Beispiel:

Der Nah-Unendlichkeitspunkt liegt bei $\bar{a} = f' \cdot \left(\frac{f'}{K \cdot u_{\max}} + 1 \right)$, sodass bei einem auf ∞ eingestellten Objektiv von der Entfernung \bar{a} an alles scharf ist. Fokussiert man das Objektiv auf die Entfernung \bar{a} (d. h., $a_0 = \bar{a}$), dann wird im Bereich $[\frac{\bar{a}}{2}, \infty]$ alles scharf.

Dabei sollte man unbedingt darauf achten, dass die Höhe der einzelnen Elemente im Text nicht zu groß wird.

Häufiger Fehler: Im Fließtext wird bei einfachen Variablen oft auf die Verwendung der richtigen, mathematischen Zeichen vergessen, wie etwa in "X-Achse" anstelle von " X -Achse" (`X -Achse`).

5.2 Freigestellte Ausdrücke

Freigestellte mathematische Ausdrücke können in LaTeX im einfachsten Fall durch paarweise `$$... $$` erzeugt werden. Das Ergebnis wird zentriert, erhält jedoch keine Nummerierung. So ist z. B.

$$y = 4x^2$$

das Ergebnis von `$$ y = 4 x^2 $$`.

5.2.1 Einfache Gleichungen

Meistens verwendet man in solchen Fällen jedoch die `equation`-Umgebung zur Herstellung numerierter Gleichungen, auf die man im Text jederzeit verweisen kann. Zum Beispiel

erzeugt

```
1 \begin{equation}
2   f(k) = \frac{1}{N} \sum_{i=0}^{k-1} i^2 .
3   \label{eq:MyFirstEquation}
4 \end{equation}
```

die Gleichung

$$f(k) = \frac{1}{N} \sum_{i=0}^{k-1} i^2. \quad (5.1)$$

Mit `\ref{eq:MyFirstEquation}` erhält man wie üblich die Nummer (??) dieser Gleichung (siehe dazu auch Abschn. ??). Dieselbe Gleichung *ohne* Numerierung kann man übrigens mit der `equation*`-Umgebung erzeugen.

Man beachte, dass **Gleichungen** inhaltlich ein **Teil des Texts** sind und daher neben der sprachliche **Überleitung** auch die **Punktuation** (wie in Gl. ?? gezeigt) beachtet werden muss. Bei Unsicherheiten sollte man sich passende Beispiele in einem guten Mathematikbuch ansehen.

Für Interessierte findet sich mehr zum Thema Mathematik und Prosa in [?] und [?].

5.2.2 Mehrzeilige Gleichungen

Für mehrzeilige Gleichungen bietet LaTeX die `eqnarray`-Umgebung, die allerdings etwas eigenwillige Zwischenräume erzeugt. Es empfiehlt sich, dafür gleich auf die erweiterten Möglichkeiten des `amsmath`-Pakets¹ [?] zurückzugreifen. Hier ein Beispiel mit zwei am =-Zeichen ausgerichteten Gleichungen,

$$f_1(x, y) = \frac{1}{1-x} + y, \quad (5.2)$$

$$f_2(x, y) = \frac{1}{1+y} - x, \quad (5.3)$$

erzeugt mit der `align`-Umgebung aus dem `amsmath`-Paket:

```
1 \begin{align}
2   f_1(x,y) &= \frac{1}{1-x} + y , \label{eq:f1} \\
3   f_2(x,y) &= \frac{1}{1+y} - x , \label{eq:f2}
4 \end{align}
```

5.2.3 Fallunterscheidungen

Mit der `cases`-Umgebung aus `amsmath` sind Fallunterscheidungen, u. a. innerhalb von Funktionsdefinitionen, sehr einfach zu bewerkstelligen. Beispielsweise wurde die rekursive Definition

$$H(i) = \begin{cases} h(0) & \text{für } i = 0, \\ H(i-1) + h(i) & \text{für } 0 < i \leq K. \end{cases} \quad (5.4)$$

mit folgenden Anweisungen erzeugt:

¹American Mathematical Society (AMS). `amsmath` ist Teil der LaTeX Standardinstallation und wird von `htl.sty` bereits importiert.

```

1 \begin{equation}
2 \mathsf{H}(i) =
3 \begin{cases}
4 \mathsf{h}(0) & \& \text{f"ur } \$i = 0$,\\
5 \mathsf{H}(i-1) + \mathsf{h}(i) & \& \text{f"ur } \$0 < i \leq K$.
6 \end{cases}
7 \end{equation}

```

Man beachte dabei die Verwendung des sehr praktischen `\text{...}`-Makros, mit dem im Mathematik-Modus gewöhnlicher Text eingefügt werden kann, sowie wiederum die Punctuation innerhalb der Gleichung.

5.2.4 Gleichungen mit Matrizen

Auch hier bietet `amsmath` einige Vorteile gegenüber der Verwendung der LaTeX Standardkonstrukte. Dazu ein einfaches Beispiel für die Verwendung der `pmatrix`-Umgebung für Vektoren und Matrizen,

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix}, \quad (5.5)$$

das mit den folgenden Anweisungen erzeugt wurde:

```

1 \begin{equation}
2 \begin{pmatrix}
3 x' \\
4 y'
5 \end{pmatrix}
6 =
7 \begin{pmatrix}
8 \cos \phi & -\sin \phi \\
9 \sin \phi & \phantom{-}\cos \phi
10 \end{pmatrix}
11 \cdot
12 \begin{pmatrix}
13 x \\
14 y
15 \end{pmatrix} ,
16 \end{equation}

```

Ein nützliches Detail darin ist das TeX-Makro `` (in Zeile ??), das sein Argument unsichtbar einfügt und hier als Platzhalter für das darüberliegende Minuszeichen verwendet wird. Alternativ kann man mit der `bmatrix`-Umgebung Matrizen mit eckigen Klammern erzeugen. Zahlreiche weitere mathematische Konstrukte des `amsmath`-Pakets sind in [?] beschrieben.

5.2.5 Verweise auf Gleichungen

Beim Verweis auf nummerierte Formeln und Gleichungen genügt grundsätzlich die Angabe der entsprechenden Nummer in runden Klammern, z. B.

"...wie aus (??) abgeleitet werden kann ..."

Um Missverständnisse zu vermeiden, sollte man aber – vor allem in Texten mit nur wenigen mathematischen Elementen – "Gleichung ??", "Gl. ??", oder "Gl. (??)" schreiben (natürlich konsistent).

Achtung: Vorwärtsverweise auf (im Text weiter hinten liegende) Gleichungen sind **äußerst ungewöhnlich** und sollten vermieden werden! Glaubt man dennoch so etwas zu benötigen, dann wurde meistens ein Fehler in der Anordnung gemacht.

5.3 Spezielle Symbole

5.3.1 Zahlenmengen

Einige häufig verwendete Symbole sind leider im ursprünglichen mathematischen Zeichensatz von LaTeX nicht enthalten, z. B. die Symbole für die reellen und natürlichen Zahlen. Im `htl`-Paket sind diese Symbole als Makros `\R` (\mathbb{R}), `\Z` (\mathbb{Z}), `\N` (\mathbb{N}), `\C` (\mathbb{C}) und `\Q` (\mathbb{Q}) mithilfe der *AMS Blackboard Fonts* definiert, z. B.:

$$x \in \mathbb{R}, i \in \mathbb{N}_0, z \in \mathbb{C}.$$

5.3.2 Operatoren

In LaTeX sind Dutzende von mathematischen Operatoren für spezielle Anwendungen definiert. Am häufigsten werden natürlich die arithmetischen Operatoren $+$, $-$, \cdot und $/$ benötigt. Ein dabei oft beobachteter Fehler (der wohl aus der Programmierpraxis resultiert) ist die Verwendung von `*` für die einfache Multiplikation – richtig ist `\cdot` (`\cdot`).² Für Angaben wie z. B. "ein Feld mit 25×70 Metern" (aber auch fast *nur* dafür) verwendet man sinnvollerweise den `\times` (`\times`) Operator und *nicht* einfach den Buchstaben "x".

5.3.3 Symbole mit mehreren Zeichen

Vor allem bei der mathematischen Spezifikation von Algorithmen und Programmen ist es häufig notwendig, Symbole (Variablenamen) mit mehr als einem Zeichen zu verwenden, z. B.

$$Scalefactor \leftarrow Scalefactor^2 \cdot 1.5,$$

erzeugt durch `$Scalefactor \leftarrow Scalefactor^2 \cdot 1.5$`. Dabei interpretiert LaTeX allerdings die Zeichenkette "Scalefactor" als 11 einzelne, aufeinanderfolgende Symbole S , c , a , l , e , ... und setzt dazwischen entsprechende Abstände. Richtig ist, diese Buchstaben mit `\mathit{...}` zu *einem* Symbol zusammenzufassen. Der Unterschied ist in diesem Fall deutlich sichtbar:

$$\begin{array}{ll} \text{Falsch: } Scalefactor^2 \leftarrow \$Scalefactor^2\$ \\ \text{Richtig: } Scalefactor^2 \leftarrow \$\mathit{Scalefactor}^2\$ \end{array}$$

Grundsätzlich sollte man aber derart lange Symbolnamen ohnehin vermeiden und stattdessen möglichst einzelne Symbole verwenden (z. B. Brennweite $f = 50$ mm statt *Brennweite* = 50 mm).

5.3.4 Funktionen

Während Symbole für Variablen traditionell (und in LaTeX automatisch) *italic* gesetzt werden, verwendet man für die Namen von Funktionen und Operatoren üblicherweise *roman* als Schrifttyp, wie z. B. in

$$\sin \theta = \sin(\theta + 2\pi) \quad \$\sin \theta = \sin(\theta + 2 \pi)$$$

²Das Zeichen `*` wird üblicherweise für den Faltungsoperator verwendet.

Das ist bei den bereits vordefinierten Standardfunktionen (wie `\sin`, `\cos`, `\tan`, `\log`, `\max` u. v. a.) automatisch der Fall. Diese Konvention sollte man auch bei selbstdefinierten Funktionen befolgen, wie etwa in

$$\text{Distance}(A, B) = |A - B| \quad \$\mathrm{Distance}(A, B) = |A - B| \$$$

5.3.5 Maßeinheiten und Währungen

Bei der Angabe von Maßeinheiten wird üblicherweise Normalschrift (keine *Italics*) verwendet, z. B.:

Die Höchstgeschwindigkeit der *Bell XS-1* beträgt 345 m/s bei einem Startgewicht von 15 t. Der Prototyp kostete über 25 000 000 US\$, also ca. 19 200 000 € nach heutiger Umrechnung.

Der Abstand zwischen der Zahl und der Maßeinheit ist dabei gewollt. Das \$-Zeichen erzeugt man mit `\$`, während das Euro-Symbol (€) nicht im ursprünglichen LaTeX-Zeichensatz enthalten ist. Es wird durch das Makro `\euro` zur Verfügung gestellt. Die kleinen Abstände zwischen den Ziffernblöcken der Beträge wurden hier übrigens mit `\thinspace` erzeugt.

5.3.6 Kommas in Dezimalzahlen (Mathematik-Modus)

LaTeX setzt im Mathematik-Modus (also innerhalb von `$$` oder in Gleichungen) nach dem anglo-merikanischen Stil in Dezimalzahlen grundsätzlich den *Punkt* (.) als Trennsymbol voraus. So wird etwa mit `3.141` normalerweise die Ausgabe "3.141" erzeugt. Um das in Europa übliche Komma in Dezimalzahlen zu verwenden, genügt es *nicht*, einfach . durch , zu ersetzen. Letzteres würde als Satzzeichen interpretiert und sieht dann so aus:

$$\$3,141\$ \rightarrow 3,141$$

(beachte den Leerraum nach dem Komma). Dieses Verhalten lässt sich in LaTeX zwar global umdefinieren, was aber wiederum zu einer Reihe unangenehmer Nebeneffekte führt. Eine einfache (wenn auch nicht sehr elegante) Lösung ist, Kommazahlen im Mathematik-Modus so zu schreiben:

$$\$3\{, \}141\$ \rightarrow 3,141$$

5.3.7 Mathematik-Werkzeuge

Für die Erstellung komplizierter Gleichungen ist es mitunter hilfreich, auf spezielle Software zurückzugreifen. Unter anderem kann man aus dem Microsoft *Equation Editor* und aus *Mathematica* auf relativ einfache Weise LaTeX-Anweisungen für mathematische Gleichungen exportieren und direkt (mit etwas manueller Nacharbeit) in das eigene LaTeX-Dokument übernehmen.

5.4 Algorithmen

Für die Beschreibung von Algorithmen in mathematischer Form oder Pseudocode ist in LaTeX selbst keine spezielle Unterstützung vorgesehen. Dazu gibt es jedoch eine Reihe von LaTeX-Paketen (z. B. `algorithms`, `algorithmicx`, `algorithm2e`). Das Beispiel in Alg. ??

wurde mit der Float-Umgebung von `algorithm` und dem algorithmischen Schriftsatz aus `algorithmicx` ausgeführt.³ Die grobe Struktur dieses Konstrukts ist

```

1 \begin{algorithm}
2 \caption{Bikubische Interpolation in 2D.}
3 \label{alg:xyz}
4 \begin{algorithmic}[1]
5   \Procedure{BicubicInterpolation}{$I, x_0, y_0$}
6     ...
7   \EndProcedure
8 \end{algorithmic}
9 \end{algorithm}

```

Weitere Details finden sich im Quelltext und natürlich in der Dokumentation der verwendeten Pakete. Umfangreichere Beispiele für Algorithmen mit diesem Setup findet man u. a. in [?].

Algorithm 5.1: Bikubische Interpolation in 2D.

```

1: BICUBICINTERPOLATION( $I, x_0, y_0$ ) ▷  $x_0, y_0 \in \mathbb{R}$ 
   Returns the interpolated value of the image  $I$  at the continuous position  $(x_0, y_0)$ .
2:    $q \leftarrow 0$ 
3:   for  $j \leftarrow 0 \dots 3$  do ▷ iterate over 4 lines
4:      $v \leftarrow \lfloor y_0 \rfloor - 1 + j$ 
5:      $p \leftarrow 0$ 
6:     for  $i \leftarrow 0 \dots 3$  do ▷ iterate over 4 columns
7:        $u \leftarrow \lfloor x_0 \rfloor - 1 + i$ 
8:        $p \leftarrow p + I(u, v) \cdot w_{\text{cub}}(x_0 - u)$ 
9:     end for
10:     $q \leftarrow q + p \cdot w_{\text{cub}}(y_0 - v)$ 
11:  end for
12:  return  $q$ .
13: end

```

³Hier wird `algorithmicx` bzw. `algpseudocode` in der älteren Version 1.1 verwendet (die zugehörigen `.sty`-Dateien liegen dieser Vorlage bei), weil die aktuelle Version (1.2) offenbar fehlerhaft ist. Als Alternative sollte man sich eventuell das `algorithm2e`-Paket ansehen.

Chapter 6

Umgang mit Literatur und anderen Quellen

Anmerkung: Der Titel dieses Kapitels ist für die Kopfzeile (absichtlich) zu lang geraten; in diesem Fall kann man in der `chapter`-Anweisung von LaTeX als optionales Argument [...] einen verkürzten Text für die Kopfzeile angeben:

```
\chapter[Umgang mit Literatur]
      {Umgang mit Literatur und anderen Quellen}
```

6.1 Allgemeines

Für die Gestaltung der Literaturverweise im Text und der Quellenangaben sind weltweit eine Vielzahl verschiedener Richtlinien in Gebrauch. Die Wahl des richtigen Schemas ist Geschmacksache – wichtig ist jedoch, eine durchdachte und vor allem *konsistente* Verwendung. Das Literaturverzeichnis ist eine Zusammenstellung der verwendeten Quellen am *Ende* des Dokuments. Wichtig ist, dass jeder Literaturverweis im Text einen entsprechenden Eintrag im Literaturverzeichnis hat und umgekehrt.

6.2 Finden von Literatur

6.3 Literaturverweise

6.3.1 cite

Für Literaturverweise im laufenden Text verwendet man in LaTeX die Anweisung

`\cite{Verweise}` oder `\cite[Zusatztext]{Verweise}`.

Verweise ist eine durch Kommas getrennte Auflistung von Quellen-Schlüsseln zur Identifikation der entsprechenden Einträge im Literaturverzeichnis. Mit *Zusatztext* können Ergänzungstexte zum aktuellen Literaturverweis angegeben werden, wie z. B. Kapitel- oder Seitenangaben bei Büchern. Einige Beispiele dazu:

Mehr dazu findet sich in `\cite{Kopka98}`.

→ Mehr dazu findet sich in [?].

Mehr über `\emph{Styles}` in `\cite[Kap.\ 3]{Kopka98}`.

→ Mehr über *Styles* in [?, Kap. 3].

Die Angaben in `\cite[S.\ 274--277]{Ears99}` sind falsch.

→ Die Angaben in [?, S. 274–277] sind aus meiner Sicht falsch.

überholt sind auch `\cite{Ears99,Microsoft01,Duden97}`.
 → überholt sind auch `[?, ?, ?]`.

Die im letzten Beispiel sichtbare automatische Sortierung der Angaben in der Verweisliste wird übrigens durch die Verwendung des `cite`-Pakets erreicht.

6.3.2 Häufige Fehler

Verweise außerhalb des Satzes

Literaturverweise sollten innerhalb oder am Ende eines Satzes (vor dem Punkt) stehen, nicht *außerhalb*, wie hier. `[?] ← FALSCH!`

Zitate

Falls ein ganzer Absatz (oder mehr) aus einer Quelle zitiert wird, sollte man den Verweis im vorlaufenden Text platzieren und nicht *innerhalb* des Zitats selbst. Zum Beispiel, die folgenden klaren Worte aus `[?]`:

Typographical design is a craft. Unskilled authors often commit serious formatting errors by assuming that book design is mostly a question of aesthetics—"If a document looks good artistically, it is well designed." But as a document has to be read and not hung up in a picture gallery, the readability and understandability is of much greater importance than the beautiful look of it.

Für das Zitat selbst sollte man übrigens das dafür vorgesehene

`\begin{quote} ... \end{quote}`

Environment verwenden, das durch beidseitige Einrückungen das Zitat vom eigenen Text klar abgrenzt und damit die Gefahr von Unklarheiten (wo ist das Ende des Zitats?) mindert. Wenn man möchte, kann man das Innere des Zitats auch in Hochkommas verpacken oder kursiv setzen – aber nicht beides!

6.4 Plagiarismus

Als "Plagiat" bezeichnet man die Darstellung eines fremden Werks als eigene Schöpfung, in Teilen oder als Ganzes, egal ob bewusst oder unbewusst. Plagiarismus ist kein neues Problem im Schulwesen, hat sich aber durch die breite Verfügbarkeit elektronischer Quellen in den letzten Jahren dramatisch verstärkt und wird keineswegs als Kavaliersdelikt betrachtet. Viele Schulen bedienen sich als Gegenmaßnahme heute ebenfalls elektronischer Hilfsmittel (die den Schülern zum Teil nicht zugänglich sind), und man sollte daher bei jeder abgegebenen Arbeit damit rechnen, dass sie routinemäßig auf Plagiatsstellen untersucht wird! Werden solche erst zu einem späteren Zeitpunkt entdeckt, kann das im schlimmsten Fall sogar zur nachträglichen (und endgültigen) Aberkennung des Abschlusses führen.

Um derartige Probleme zu vermeiden, sollte man eher übervorsichtig agieren und zumindest folgende Regeln beachten:

- Die Übernahme kurzer Textpassagen ist nur unter korrekter Quellenangabe zulässig, wobei der Umfang (Beginn und Ende) des Textzitats in jedem einzelnen Fall klar erkenntlich gemacht werden muss.
- Insbesondere ist es nicht zulässig, eine Quelle nur eingangs zu erwähnen und nachfolgend wiederholt nicht-ausgezeichnete Textpassagen als eigene Wortschöpfung zu übernehmen.

- Auf gar keinen Fall tolerierbar ist die direkte oder paraphrierte Übernahme längerer Textpassagen, ob mit oder ohne Quellenangabe. Auch indirekt übernommene oder aus einer anderen Sprache übersetzte Passagen müssen mit entsprechenden Quellenangaben gekennzeichnet sein!

Im Zweifelsfall findet man detailliertere Regeln in jedem guten Buch über wissenschaftliches Arbeiten oder man fragt sicherheitshalber den Betreuer der Arbeit.

6.5 Literaturverzeichnis

Für die Erstellung des Literaturverzeichnisses gibt es in LaTeX zwei Möglichkeiten:

1. Das Literaturverzeichnis manuell zu formatieren (s. [?, S. 56–57]).
2. Die Verwendung von BibTeX und einer zugehörigen Literaturdatenbank (s. [?, S. 245–255]).

Tatsächlich ist die erste Variante nur bei sehr wenigen Literaturangaben interessant. Die Arbeit mit BibTeX macht sich hingegen schnell bezahlt und ist zudem wesentlich flexibler.

6.5.1 Literaturdaten in BibTeX

BibTeX ist ein eigenständiges Programm, das aus einer "Literaturdatenbank" (eine oder mehrere Textdateien von vorgegebener Struktur) ein für LaTeX geeignetes Literaturverzeichnis erzeugt. Dabei ist es möglich, aus einer Reihe von verschiedenen Stilvarianten (*bibliography styles*) zu wählen. Literatur zur Verwendung von BibTeX findet man online, z. B. in [?, ?].

Man kann BibTeX-Dateien natürlich mit einem Texteditor manuell erstellen, für viele Literaturquellen sind auch bereits fertige BibTeX-Einträge im Web verfügbar. Darüber hinaus gibt es einige Software-Werkzeuge zur Wartung von BibTeX-Verzeichnissen, zu empfehlen ist beispielsweise JabRef.¹

In diesem Dokument wird der BibTeX-Stil **bababrv** ("Babel abbreviated"),² eine Variante der Originalversion **abbrv**, verwendet. Eine weitere häufig verwendete Stilvariante ist **babplain** (bzw. **plain**). Im LaTeX-Quelltext wird das Literaturverzeichnis am Ende des Dokuments in dieser Form eingesetzt:

```
\bibliographystyle{bababrv}
\bibliography{literatur}
```

Die zweite Zeile verweist auf die Literaturdatenbank in der Datei **literatur.bib**, der von BibTeX verarbeitet wird. Unter der Annahme, dass die Hauptdatei der Arbeit **da.tex** heißt, sind zur Erzeugung der Literaturliste folgende Befehle nötig (im "Command Prompt"):

```
> latex da
> bibtex da
> latex da
> latex da
```

In der *TeXnicCenter*-Umgebung wird (bei richtiger Einstellung) die erforderliche BibTeX-Anweisungsfolge automatisch bei jedem LaTeX-Durchlauf ausgeführt.

¹<http://jabref.sourceforge.net>

²Aus dem **babelbib**-Paket von Harald Harders (<http://www.ctan.org/tex-archive/biblio/bibtex/contrib/babelbib/>, s. auch Abschn. ??).

6.5.2 Beispiele

Im Folgenden einige Beispiele für die wichtigsten Formen von Quellenangaben und die zugehörigen Einträge in der BibTeX-Datei. Die Form der Literaturangaben ist natürlich abhängig vom verwendeten BibTeX-Stil, für die hier gezeigten Beispiele wurde `bababrv` verwendet. Weitere Beispiele finden sich im übrigen Text bzw. im Literaturverzeichnis.

Buch (@book)

Elemente:

Autor(en), Buchtitel, Verlag, Erscheinungsort, Erscheinungsjahr.

Beispiel:

BURGER, W. and M. BURGE: *Digitale Bildverarbeitung – Eine Einführung mit Java und ImageJ*. Springer-Verlag, Heidelberg, 2. ed., 2005.

BibTeX-Eintrag:

```

1 @book{BurgerBurge06,
2   author={Burger, Wilhelm and Burge, Mark},
3   title={Digitale Bildverarbeitung --
4     Eine Einführung mit Java und ImageJ},
5   publisher={Springer-Verlag},
6   address={Heidelberg},
7   edition={2},
8   year={2005},
9   language={german}
10 }
```

Buchbeitrag, Beitrag in einem Sammelband (@incollection)

Elemente:

Autor(en), Buchtitel, Herausgeber, Buchtitel, [Kapitel,] Verlag, Erscheinungsort, Erscheinungsjahr.

Beispiel:

BURGE, M. and W. BURGER: *Ear Biometrics*. In JAIN, A. K., B. RUUD and P. SHARATH (eds.): *Biometrics: Personal Identification in Networked Society*, ch. 13. Kluwer Academic Publishers, Boston, 1999.

BibTeX-Eintrag:

```

1 @incollection{Ears99,
2   author={Burge, Mark and Burger, Wilhelm},
3   title={Ear Biometrics},
4   booktitle={Biometrics: Personal Identification in Networked
5     Society},
6   publisher={Kluwer Academic Publishers},
7   year={1999},
8   address={Boston},
9   editor={Jain, Anil K. and Bolle Ruud and Pankanti Sharath},
10  chapter={13},
11  language={english}
12 }
```

Konferenzbeitrag, Beitrag in einem Tagungsband (@inproceedings)

Elemente:

Autor(en), Titel des Beitrags, Titel des Konferenzbands (*Conference Proceedings*),

Seiten, Ort der Tagung, Monat/Jahr der Tagung, Verlag.

Beispiel:

BURGER, W. and B. BHANU: *Qualitative Motion Understanding*. In *Proceedings of the Intl. Joint Conference on Artificial Intelligence*, pp. 819–821, San Francisco, May 1987. Morgan Kaufmann Publishers.

BibTeX-Eintrag:

```
1 @inproceedings{Burger87,
2   author={Burger, Wilhelm and Bhanu, Bir},
3   title={Qualitative Motion Understanding},
4   booktitle={Proceedings of the Intl.\ Joint Conference
5             on Artificial Intelligence},
6   year={1987},
7   month=MAY,
8   publisher={Morgan Kaufmann Publishers},
9   address={San Francisco},
10  pages={819-821},
11  language={english}
12 }
```

Zeitschriftenbeitrag, Journal Paper (@article)

Elemente:

Autor(en), Titel des Beitrags, Zeitschrift, Band, Seiten, Monat(e), Erscheinungsjahr.

Beispiel:

GUTTMAN, E.: *Autoconfiguration for IP Networking*. IEEE Internet Computing, 5:81–86, Mai/Juni 2001.

BibTeX-Eintrag:

```
1 @article{Guttman01,
2   author={Guttman, Erik},
3   title={Autoconfiguration for {IP} Networking},
4   journal={IEEE Internet Computing},
5   volume={5},
6   year={2001},
7   pages={81-86},
8   month=MAY # "/" # JUN,
9   language={english}
10 }
```

Dissertation (@phdthesis)

Elemente:

Autor, Titel, Hochschule, Institut, Adresse, Monat, Jahr.

Beispiel:

EBERL, G.: *Automatischer Landeanflug durch Rechnersehen*. PhD thesis, Universität der Bundeswehr, Fakultät für Raum- und Luftfahrttechnik, München, August 1987.

BibTeX-Eintrag:

```
1 @phdthesis{Eberl87,
2   author={Eberl, Gerhard},
3   title={Automatischer Landeanflug durch Rechnersehen},
4   school={Universität der Bundeswehr, Fakultät für
5          Raum- und Luftfahrttechnik},
6   year={1987},
7   month=AUG,
```

```

8   address={München},
9   language={german}
10 }

```

Diplomarbeit (@mastersthesis)

Elemente:

Autor, Titel, Hochschule, Institut, Adresse, Monat, Jahr.

Beispiel:

WINTERSBERGER, M.: *Realisierung eines Internet-Auktionshauses basierend auf PHP und MySQL*. Master's thesis, Fachhochschule Hagenberg, Medientechnik und -design, Hagenberg, Austria, Juni 2000.

BibTeX-Eintrag:

```

1 @mastersthesis{Wintersberger00,
2   author={Wintersberger, Markus},
3   title={Realisierung eines Internet-Auktionshauses
4         basierend auf {PHP} und {MySQL}},
5   school={Fachhochschule Hagenberg, Medientechnik und -design},
6   year={2000},
7   month=JUN,
8   address={Hagenberg, Austria},
9   language={german}
10 }

```

Bachelorarbeit (@masterthesis modifiziert)

Bachelorarbeiten gelten in der Regel zwar nicht als "richtige" Publikationen, bei Bedarf muss man sie aber doch referenzieren können. Das geht ebenfalls mit dem Eintrag `@mastersthesis{}`, versehen mit einem entsprechenden `type`-Feld, zum Beispiel [?]:

BibTeX-Eintrag:

```

1 @mastersthesis{Bacher04,
2   author={Bacher, Florian},
3   title={Interaktionsmöglichkeiten mit Bildschirmen
4         und gro\ss flächigen Projektionen},
5   school={Fachhochschule Hagenberg, Medientechnik und -design},
6   year={2004},
7   month=JUN,
8   address={Hagenberg, Austria},
9   type={Bachelorarbeit},
10  url={http://theses.fh-hagenberg.at},
11  language={german}
12 }

```

Man beachte, dass der Inhalt des Eintrags `url={..}` automatisch und ohne weitere Auszeichnung mit den Einstellungen des `\url{}` Makros gesetzt wird.

Bericht, Technical Report (@techreport)

Das sind typischerweise nummerierte Berichte aus Unternehmen, Hochschulinstituten oder Forschungsprojekten, Wichtig ist, dass die herausgebende Organisationseinheit (Institut, Fakultät etc.) und Adresse vollständig angegeben wird.

Elemente:

Autor(en), Titel, Report-Nummer, Institution, Adresse, Monat, Jahr.

Beispiel:

BEELER, D. E. and J. P. MAYER: *Measurement of the wing and tail loads during acceptance test of the Bell XS-1 research airplane*. Tech. Rep. NACA-RM-L7L12, NASA Dryden Flight Research Center, Edwards, CA, April 1948. www.dfrc.nasa.gov/DTRS/1948/index.html

BibTeX-Eintrag:

```
1 @techreport{Beeler48,
2   author={Beeler, De E. and Mayer, John P.},
3   title={Measurement of the wing and tail loads
4         during acceptance test of the {Bell}
5         {XS-1} research airplane},
6   institution={NASA Dryden Flight Research Center},
7   year={1948},
8   address={Edwards, CA},
9   number={NACA-RM-L7L12},
10  url={www.dfrc.nasa.gov/DTRS/1948/index.html},
11  language={english}
12 }
```

Handbuch, Manual (@manual)

Dieser Publikationstyp bietet sich zum einen natürlich für Handbücher und Produktbeschreibungen an, insbesondere aber auch für viele Dokumente, bei denen etwa kein Autor genannt ist, wie bei vielen Online-Quellen.

Elemente:

Autor(en) oder Institution, Titel des Dokuments, Adresse, Datum.

Beispiel:

MICROSOFT CORPORATION: *Application Service Providers: Evolution and Resources*, Jan. 2001. www.microsoft.com/ISN/downloads/ASP.doc, Kopie auf CD-ROM.

BibTeX-Eintrag:

```
1 @manual{Microsoft01,
2   organization={Microsoft Corporation},
3   title={Application Service Providers: Evolution and Resources},
4   year={2001},
5   month=JAN,
6   url={www.microsoft.com/ISN/downloads/ASP.doc},
7   note={Kopie auf CD-ROM},
8   language={english}
9 }
```

Sonderfälle (@misc)

Es kommt immer wieder vor, dass spezielle Quellen zu berücksichtigen sind. Bei Technischen Diplomarbeiten fallen vor allem Normen und Patente in diesen Bereich. Andere Beispiele sind Filme, Videos oder Musikstücke. Für diese gibt es in BibTeX leider keine eigenen Vorkehrungen, sie können aber mit dem @misc-Eintrag erstellt werden. Nachfolgend einige Beispiele für häufig erforderliche Situationen.

Patent: In Patenten befindet sich ein großer Teil des aktuellsten Technischen Wissens. Dementsprechend sind in technischen Diplomarbeiten häufig Patente als Quellen anzuführen. Als Beispiel dient hier das European Patent No. EP1765031 [?]

BibTeX-Eintrag:


```

1 @misc {PAT07,
2   author = {M. Anisetti and {C.A.} Ardagna and V. Bellandi and E. Damiani},
3   howpublished = {European Patent No. EP1765031},
4   title = {Method, System, Network and Computer Program Product for
5     Positioning in a Mobile Communications Network},
6   month = MAR,
7   day = {21},
8   year = {2007},
9   language={english}
10 }

```

Norm: Normen sind in der Technik allgegenwärtig. Beispiele sind zwei Teile der Norm DIN 9241. Teil 11[?] und Teil 110[?].

BibTeX-Eintrag:

```

1 @misc{DIN9241-110,
2   publisher = {Europäisches Komitee für Normung},
3   title = {Ergonomie der Mensch-System-Interaktion: Teil 110 Grundsätze der
4     Dialoggestaltung},
5   howpublished = {DIN 9241-110},
6   year = 2006,
7   language={german}
8 }
9 @misc{DIN9241-11,
10  howpublished = {DIN EN ISO 9241-11},
11  title = {Ergonomische Anforderungen für Bildschirmtextarbeiten
12    Teil 11: Anforderungen an die Gebrauchstauglichkeit - Leitsätze},
13  year = 1999,
14  month =JAN,
15  language={german}
16 }

```

Audio-CD: Hier ein Vorschlag unter der Annahme, dass es sich um das Werk eines oder mehrerer klar identifizierbarer Künstler (Autoren) handelt. Ansonsten könnte man alternativ ein Format ähnlich wie für einen Film (s. unten) verwenden.

Beispiel:

ZAPPA, F.: *Freak Out*. Audio-CD, Rykodisc, New York, May 1995.

BibTeX-Eintrag:

```

1 @misc{Zappa95,
2   author={Zappa, Frank},
3   title={Freak Out},
4   year={1995},
5   month=MAY,
6   howpublished={Audio-CD, Rykodisc, New York},
7   language={english}
8 }

```

Film/DVD: In diesem Fall wird bewusst *kein* Autor angegeben, weil dieser bei einem Film i. Allg. nicht eindeutig zu benennen ist. Aus diesem Grund wird ein **key**-Feld angegeben, das (anstelle des Autors) für die Sortierung verwendet wird.

Beispiel:

Nosferatu – A Symphony of Horrors. Drehbuch/Regie: F. W. Murnau. Mit Max

Schreck, Gustav von Wangenheim, Greta Schröder, 1922. DVD, Film Preservation Associates, London (1991).

BibTeX-Eintrag:

```
1 @misc{Nosferatu,
2   key={Nosferatu},
3   title={Nosferatu -- A Symphony of Horrors},
4   year={1922},
5   howpublished={Drehbuch/Regie: F. W. Murnau.
6   Mit Max Schreck, Gustav von Wangenheim, Greta Schröder},
7   note={DVD, Film Preservation Associates, London (1991)},
8   language={german}
9 }
```

Ein weiteres Filmbeispiel ist *Psycho* von Alfred Hitchcock [?].

Persönliche Kommunikation: Ein Sonderfall sind Quellen ohne irgendeiner Form von Veröffentlichung, wie z. B. Hinweise aus persönlichen Gesprächen (*personal communication*) mit Fachleuten. Obwohl man auf diese Art von Verweisen **nur im äußersten Fall** zurückgreifen sollte, hier ein Beispiel wie man damit umgeht:

Beispiel:

KREISKY, B.: *Kaffeehausgespräch zur Lage der Nation*. Persönl. Gespräch, Nov. 1975.

BibTeX-Eintrag:

```
1 @misc{Kreisky75,
2   author={Kreisky, Bruno},
3   title={Kaffeehausgespräch zur Lage der Nation},
4   year={1975},
5   month=NOV,
6   howpublished={Persönl. \ Gespräch},
7   language={german}
8 }
```

Online-Quellen, Wiki-Einträge etc.

Verweise auf Webseiten sollten generell nur in Ausnahmefällen verwendet werden und auch nur dann, wenn keine entsprechende andere Publikation verfügbar ist und sich an der angegebenen Adresse (URL) auch tatsächlich ein Dokument befindet. Bei Online-Quellen *ohne explizit angegebenem Autor*, z. B. Firmen-Homepages, Link-Sammlungen und vor allem auch *Wikipedia*-Seiten, sollte man die entsprechende Webadresse *nicht* in die Literaturliste aufnehmen sondern direkt im Text als **Fußnote** angeben. Beispielsweise bezeichnet man als "Reliquienschrein" einen Schrein, in dem die Reliquien eines oder mehrerer Heiliger aufbewahrt werden.³

Wird der Diplomarbeit ein elektronischer Datenträger (CD-ROM, DVD etc.) beigelegt, empfiehlt sich die angeführten Webseiten in elektronischer Form (vorzugsweise als PDF-Dateien) abzulegen und mit einem entsprechenden Verweis im Literaturverzeichnis ("Kopie auf CD-ROM") zu versehen.

³<http://de.wikipedia.org/wiki/Reliquienschrein>

6.5.3 Tipps zur Erstellung von BibTeX-Dateien

month-Attribut

Für die Angabe des `month`-Attributs sollte man grundsätzlich die zwölf in BibTeX bereits vordefinierten Abkürzungen

JAN, FEB, MAR, APR, MAY, JUN, JUL, AUG, SEP, OCT, NOV, DEC

verwenden, und zwar *ohne* die sonst erforderlichen Klammern (oder Hochkommas), also beispielsweise einfach mit

`month=AUG`

Der richtige Monatsname wird abhängig von der Spracheinstellung für das Dokument automatisch eingesetzt. Ein Intervall über *mehrere* Monate kann man in der (etwas eigenartigen) BibTeX-Syntax so angeben (verwendet z. B. für "Mai/Juni" in [?]):

`month=MAY # "/" # JUN`

language-Attribut

Das in dieser Vorlage verwendete `babelbib`-Paket ermöglicht den korrekten Satz mehrsprachiger Literaturverzeichnisse. Dazu ist es ratsam, bei jedem Quelleneintrag auch die entsprechende Sprache anzugeben, also beispielsweise

`language={german}` oder `language={english}`

für ein deutsch- bzw. englischsprachiges Dokument.

edition-Attribut

Die Verwendung von `babelbib` lässt auch die früheren Probleme mit der Angabe der Auflagennummer von Büchern. Nunmehr ist lediglich die Nummer selbst anzugeben, also etwa

`edition={3}`

bei einer dritten Auflage. Die richtige Punctuation in der Quellenangabe wird in Abhängigkeit von der Spracheinstellung durch `babelbib` automatisch hinzugefügt ("3. Auflage" bzw. "3rd edition").

Listing aller Quellen

Durch die Anweisung `\nocite{*}` – an beliebiger Stelle im Dokument platziert – werden *alle* bestehenden Einträge der BibTeX-Datei im Literaturverzeichnis aufgelistet, also auch jene, für die es keine explizite `\cite{}` Anweisung gibt. Das ist ganz nützlich, um während des Schreibens der Arbeit eine aktuelle Übersicht auszugeben. Normalerweise müssen aber alle angeführten Quellen auch im Text referenziert sein!

Chapter 7

Drucken der Diplomarbeit

7.1 PDF-Workflow

In der aktuellen Version wird LaTeX so benutzt, dass damit direkt PDF-Dokumente (ohne den früher üblichen Umweg über DVI und PS) erzeugt werden.

7.2 Drucken

7.2.1 Drucker und Papier

Die Diplomarbeit sollte in der Endfassung unbedingt auf einem qualitativ hochwertigen Laserdrucker ausgedruckt werden, Ausdrücke mit Tintenstrahldruckern sind *nicht* ausreichend. Auch das verwendete Papier sollte von guter Qualität (holzfrei) und üblicher Stärke (mind. 80 g/m²) sein. Falls *farbige* Seiten notwendig sind, sollte man diese einzeln¹ auf einem Farb-Laserdrucker ausdrucken und dem Dokument beifügen.

Übrigens sollten *alle* abzugebenden Exemplare **gedruckt** (und nicht kopiert) werden! Die Kosten für den Druck sind heute nicht höher als die für Kopien, der Qualitätsunterschied ist jedoch – vor allem bei Bildern und Grafiken – meist deutlich.

7.2.2 Druckgröße

Ein häufiger und leicht zu übersehender Fehler beim Ausdrucken von PDF-Dokumenten wird durch die versehentliche Einstellung der Option "Fit to page" im Druckmenü verursacht, wobei die Seiten meist zu klein ausgedruckt werden. überprüfen Sie daher die Größe des Ausdrucks anhand der eingestellten Zeilenlänge oder mithilfe einer Messgrafik, wie am Ende dieses Dokuments gezeigt. Sicherheitshalber sollte man diese Messgrafik bis zur Fertigstellung der Arbeit beizubehalten und die entsprechende Seite erst ganz am Schluss zu entfernen. Wenn, wie häufig der Fall, einzelne Seiten getrennt in Farbe gedruckt werden, so sollten natürlich auch diese genau auf die Einhaltung der Druckgröße kontrolliert werden!

7.3 Binden

Die Endfassung der Diplomarbeit ist in fest gebundener Form einzureichen. Dabei ist eine Bindung zu verwenden, die das Ausfallen von einzelnen Seiten nachhaltig verhindert, z. B.

¹Tip: Mit *Adobe Acrobat* lassen sich sehr einfach einzelne Seiten des Dokuments für den Farbdruck auswählen und zusammenstellen.

durch eine traditionelle Rückenbindung (Buchbinder) oder durch handelsübliche Klammerungen aus Kunststoff oder Metall. Eine einfache Leimbindung ohne Verstärkung ist jedenfalls *nicht* ausreichend.

Falls man – was sehr zu empfehlen ist – die Arbeit bei einem professionellen Buchbinder durchführen lässt, sollte man auch auf die Prägung am Buchrücken achten, die kaum zusätzliche Kosten verursacht. Üblich ist dabei die Angabe des Familiennamens des Autors und des Titels der Arbeit. Ist der Titel der Arbeit zu lang, muss man notfalls eine gekürzte Version angeben, wie z. B.:

SCHLAUMEIER · PARZ. LÖSUNGEN ZUR ALLG. PROBLEMATIK

7.4 Elektronische Datenträger (CD-R, DVD)

Speziell bei Arbeiten im Bereich der Informationstechnik (aber nicht nur dort) fallen fast immer Informationen an, wie Programme, Daten, Grafiken, Kopien von Internetseiten usw., die für eine spätere Verwendung elektronisch verfügbar sein sollten. Vernünftigerweise wird man diese Daten während der Arbeit bereits gezielt sammeln und der fertigen Arbeit auf einer CD-ROM oder DVD beilegen. Es ist außerdem sinnvoll – schon allein aus Gründen der elektronischen Archivierbarkeit – die eigene Arbeit selbst als PDF-Datei beizulegen.²

Falls ein elektronischer Datenträger (CD-ROM oder DVD) beigelegt wird, sollte man auf folgende Dinge achten:

1. Jedem abzugebenden Exemplar muss eine identische Kopie des Datenträgers beiliegen.
2. Verwenden Sie qualitativ hochwertige Rohlinge und überprüfen Sie nach der Fertigstellung die tatsächlich gespeicherten Inhalte des Datenträgers!
3. Der Datenträger sollte in eine im hinteren Umschlag eingeklebte Hülle eingefügt sein und sollte so zu entnehmen sein, dass die Hülle dabei *nicht* zerstört wird (die meisten Buchbinder haben geeignete Hüllen parat).
4. Der Datenträger muss so beschriftet sein, dass er der Diplomarbeit eindeutig zuzuordnen ist, am Besten durch ein gedrucktes Label oder sonst durch *saubere* Beschriftung mit der Hand und einem feinen, wasserfesten Stift.
5. Nützlich ist auch ein (grobes) Verzeichnis der Inhalte des Datenträgers (wie exemplarisch in Anhang ??).

²Auch Bilder und Grafiken könnten in elektronischer Form nützlich sein, die LaTeX- oder Word-Dateien sind hingegen überflüssig.

Chapter 8

Hinweise für *Word*-Benutzer¹

Wie bereits in der Einleitung erwähnt, ist *Word* für das Schreiben von umfangreicheren Werken wie Bücher und Diplomschriften nur bedingt geeignet. *Word* besitzt zwar einen immens großen Funktionsumfang, manche einfach anmutende Aufgaben erfordern aber bisweilen sehr umständliche Maßnahmen oder sind schlichtweg unmöglich. Eine unangenehme Eigenschaft ist weiters, dass Word-Dokumente gelegentlich in fehlerhafte Zustände geraten können, die man nur mehr durch Rückkehr zu einer vorher gesicherten Version (sofern vorhanden) reparieren kann. Tatsächlich scheinen sich bei den neueren (XML-basierten) Office-Versionen einige der bisherigen Schwierigkeiten noch verstärkt zu haben, z. B. die noch umständlichere (und empfindliche) Verwaltung von Formatvorlagen.

Das soll nicht heißen, dass mit entsprechender Disziplin und Detailkenntnis nicht auch mit *Word* große Dokumente sauber und erfolgreich hergestellt werden können, wie auch die Produkte mancher Sachbuchverlage zeigen. Ich persönlich würde aber dazu nicht ermutigen und habe daher die in diesem Kapitel früher zusammengefassten Hinweise für den Umgang mit *Word* entfernt.

Falls man *Word* ohnehin nur oberflächlich beherrscht, sollte man daher überlegen, ob man es nicht gleich mit LaTeX versuchen möchte. Bei durchaus vergleichbarem Lernaufwand wird sich wahrscheinlich das Ausmaß an Frustration – mit Sicherheit aber das Ergebnis – deutlich unterscheiden. Falls man von Word auf LaTeX umzusteigen möchte und zu diesem Zeitpunkt bereits umfangreiches Material in *Word* vorhanden ist, sollte man sich das Programm `rtf2latex`² ansehen, das *Rich Text Format* (RTF) in LaTeX-Dateien übersetzt.

Als professionelle WYSIWYG-Alternative bietet sich z. B. *Indesign* von *Adobe* an, das für den Satz angeblich ähnliche Algorithmen wie LaTeX verwendet. Bezüglich mathematischer Elemente, gleitender Platzierung von Abbildungen und Tabellen, sowie der Verwaltung von Literaturangaben kommt Indesign allerdings an LaTeX derzeit nicht heran.

¹Dieses Kapitel ist ein Relikt aus früheren Versionen, die umfangreiche Hinweise für die Erstellung von Diplomarbeiten mit *Word* enthielten.

²z. B. www.tex.ac.uk/tex-archive/support/rtf2latex2e/

Chapter 9

Schlussbemerkungen

An dieser Stelle sollte eine Zusammenfassung der Diplomarbeit stehen, in der auch auf den Entstehungsprozess, persönliche Erfahrungen, Probleme bei der Durchführung, Verbesserungsmöglichkeiten, mögliche Erweiterungen usw. eingegangen werden kann. War das Thema richtig gewählt, was wurde konkret erreicht, welche Punkte blieben offen und wie könnte man von hier aus weiter arbeiten?

9.1 Lesen und lesen lassen

Wenn die Arbeit fertig ist, sollten Sie diese zunächst selbst nochmals vollständig und sorgfältig durchlesen, auch wenn man vielleicht das mühsam entstandene Produkt längst nicht mehr sehen möchte. Zusätzlich ist sehr zu empfehlen, auch einer weiteren Person diese Arbeit anzutun – man wird erstaunt sein, wieviele Fehler man selbst überlesen hat.

9.2 Checkliste

Abschließend noch eine kurze Liste der wichtigsten Punkte, an denen erfahrungsgemäß die häufigsten Fehler auftreten (Tab. ??).

Table 9.1: Checkliste. Diese Punkte bilden auch die Grundlage der routinemäßigen Formbegutachtung.

- ☐ **Titelseite:** Länge des Titels (Zeilenumbrüche), Name, Abteilung, Datum.
- ☐ **Erklärung:** vollständig Unterschrift.
- ☐ **Inhaltsverzeichnis:** balanzierte Struktur, Tiefe, Länge der Überschriften.
- ☐ **Kurzfassung/Abstract:** präzise Zusammenfassung, Länge, gleiche Inhalte.
- ☐ **Überschriften:** Länge, Stil, Aussagekraft.
- ☐ **Typographie:** sauberes Schriftbild, keine "manuellen" Abstände zwischen Absätzen oder Einrückungen, keine überlangen Zeilen, Hervorhebungen, Schriftgröße, Platzierung von Fußnoten.
- ☐ **Punktuation:** Binde- und Gedankenstriche richtig gesetzt, Abstände nach Punkten (vor allem nach Abkürzungen).
- ☐ **Abbildungen:** Qualität der Grafiken und Bilder, Schriftgröße und -typ in Abbildungen, Platzierung von Abbildungen und Tabellen, Captions. Sind *alle* Abbildungen (und Tabellen) im Text referenziert?
- ☐ **Gleichungen/Formeln:** mathem. Elemente auch im Fließtext richtig gesetzt, explizite Gleichungen richtig verwendet, Verwendung von mathem. Symbolen.
- ☐ **Quellenangaben:** Zitate richtig referenziert, Seiten- oder Kapitelangaben.
- ☐ **Literaturverzeichnis:** mehrfach zitierte Quellen nur einmal angeführt, Art der Publikation muss in jedem Fall klar sein, konsistente Einträge, Online-Quellen (URLs) sauber angeführt.
- ☐ **Sonstiges:** ungültige Querverweise (??), Anhang, Papiergröße der PDF-Datei ($A4 = 8.27 \times 11.69$ Zoll), Druckgröße und -qualität.

Appendix A

Technische Informationen

A.1 Aktuelle Dateiversionen

Datum	Datei
2015/07/09	<code>thldipl</code>
2015/10/07	<code>htl.sty</code>

A.2 Details zur aktuellen Version

Das ist eine völlig überarbeitete Version der Vorlage, die `pdflatex` "native" und nicht (wie bisher) im DVI-Kompatibilitätsmodus verwendet. Der primäre Anlass für diesen Schritt war die Frage, wie man automatisch Metadaten im PDF-File ablegen kann, woraus sich allerdings eine Fülle von Änderungen ergeben haben, die in Summe die Arbeit mit LaTeX um Einiges leichter machen sollten.

Es wird nunmehr als Ausgabe *direkt* PDF erzeugt und (normalerweise) keine DVI-Datei mehr. Der "klassische" DVI-PS-PDF-Modus ist allerdings weiterhin verfügbar (und auch notwendig, wenn man mit `psfrag` arbeiten möchte oder muss).

Verwendung unter Linux

Was muss ich tun, um diese Version für meine Arbeit zu verwenden:

1. Die Pakete **texlive** und **texlive-lang-german** müssen installiert sein.
2. **Eclipse** mit der Erweiterung **Texlipse** oder ein anderes LaTeX Frontend sollte installiert sein.
3. Die Dateien `htldipl.cls` und `htl.sty` sind in das eigene Projektverzeichnis zu kopieren.

Verwendung unter Windows

Was muss ich tun, um diese Version für meine Arbeit zu verwenden:

1. **MikTeX 2.8** oder höher muss installiert sein.
2. Die Dateien `htldipl.cls` und `htl.sty` sind in das eigene Arbeitsverzeichnis zu kopieren.

Verwendung unter Mac OS

Unter Mac OS wurde die aktuelle Vorlage noch nicht getestet. Der folgende Text bezieht sich auf die alte Vorlage und ist möglicherweise nicht mehr korrekt.

Diese Version sollte insbesondere unter *MacTeX* problemlos laufen. Was ist konkret zu tun, um die aktuelle Version unter Mac OS zu verwenden:

1. *MacTeX* (2009 oder höher) muss installiert sein.
2. Ein PDF-Viewer muss verfügbar sein (z. B. *Mac OS Preview*) – *TeXworks* hat eine eigenständige PDF Ausgabe inkludiert.
3. Die Zeichenkodierung des Editors muss auf ISO-8859-1 (Latin 1) gestellt sein.
4. Die Dateien `htldipl.cls` und `htl.sty` sind in das eigene Arbeitsverzeichnis zu kopieren.

A.3 Einstellungen unter Windows

Die folgenden Angaben beziehen sich auf eine bewährte Arbeitsumgebung unter MS Windows (XP, Vista, Win7) mit MikTeX und TeXnicCenter, mit folgenden Installationspfaden:

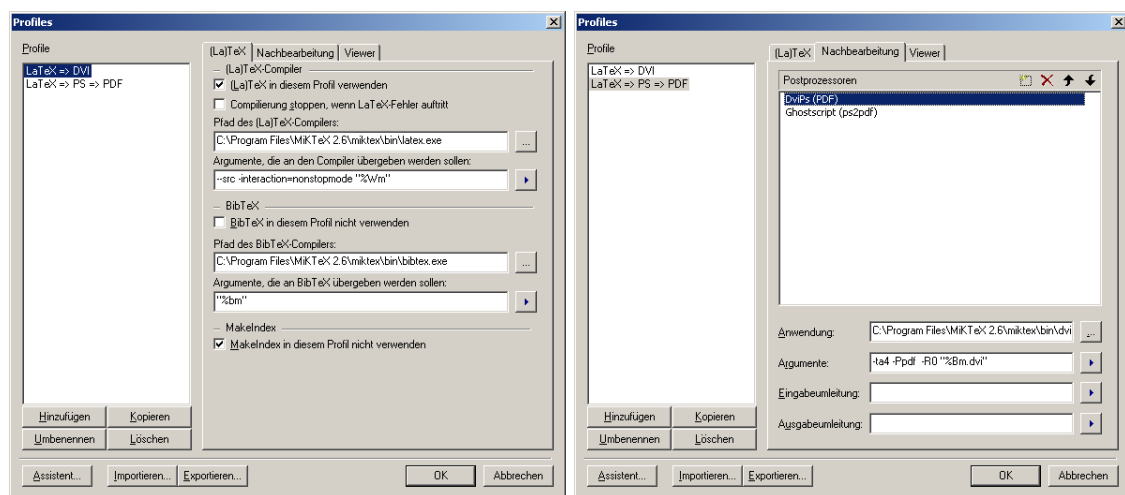
```
C:\Program Files\MikTeX 2.8\  
C:\Program Files\TeXnicCenter\
```

Falls neuere Versionen dieser Komponenten installiert sind, müssen natürlich die nachfolgend angegebenen Pfade entsprechend modifiziert werden.

A.3.1 TeXnicCenter-Ausgabeprofile

TeXnicCenter definiert den Verarbeitungsablauf des LaTeX-Dokuments anhand von Ausgabeprofilen, wobei die oben genannten Komponenten als externe Programme mit entsprechenden Argumenten aufgerufen werden. Die Einstellung der Ausgabeprofile erfolgt in TeXnicCenter über das Menü **Ausgabe**→**Ausgabeprofile definieren...** (Abb. ??). Die Profile werden (abhängig von der installierten Software) üblicherweise beim ersten Start von TeXnicCenter durch den zugehörigen "Wizard" voreingestellt.

Für diese Vorlage wird die Verwendung des Ausgabeprofils **LaTeX => PDF** empfohlen.



(a)

(b)

Figure A.1: Spezifikation der Ausgabepprofile in TeXnicCenter.

Appendix B

Inhalt der CD-ROM/DVD

Format: CD-ROM, Single Layer, ISO9660-Format¹

B.1 PDF-Dateien

Pfad: /

[_Diplomarbeit.pdf](#) . . . Diplomarbeit mit Instruktionen (Gesamtdokument)

B.2 LaTeX-Dateien

Achtung: Die folgende Auflistung soll nur den Gebrauch dieser Vorlage erleichtern. Es ist bei einer Diplomarbeit i. Allg. *nicht* notwendig, die zugehörigen LaTeX-Dateien aufzulisten (wohl aber projektbezogene Dateien, Ergebnisse, Bilder, Kopien von Online-Literatur etc.)!

Pfad: /

[_Diplomarbeit.tex](#) . . . Diplomarbeit (Hauptdokument)
[vorwort.tex](#) Vorwort
[kurzfassung.tex](#) Kurzfassung
[abstract.tex](#) Abstract
[einleitung.tex](#) Kapitel 1
[diplomschrift.tex](#) Kapitel 2
[latex.tex](#) Kapitel 3
[abbildungen.tex](#) Kapitel 4
[formeln.tex](#) Kapitel 5
[literatur.tex](#) Kapitel 6
[drucken.tex](#) Kapitel 7
[word.tex](#) Kapitel 8
[schluss.tex](#) Kapitel 9
[anhang_a.tex](#) Anhang A (Source Code)
[anhang_b.tex](#) Anhang B (Inhalt CD-ROM)
[anhang_c.tex](#) Anhang C (Änderungen)

¹Verwenden Sie möglichst ein Standardformat, bei DVDs natürlich eine entsprechende andere Spezifikation.

anhang_d.tex	Anhang D (Latex Quellcode)
messbox.tex	Messbox zur Druckkontrolle
literatur.bib	Literatur-Datenbank (BibTeX-File)

B.3 Style/Class-Dateien

Pfad: /

htldipl.sty	Class-File für dieses Dokument
htl.sty	Style-File für dieses Dokument
algorithmicx.sty	algorithmicx Paket (Version 1.1)
algpseudocode.sty	algpseudocode Paket
listings2.cfg	listings2 Paket
listings2.sty	listings2 Paket
lstaspects.sty	listings2 Paket
lstdoc2.sty	listings2 Paket
lstlanguages.sty	listings2 Paket

B.4 Sonstiges

Pfad: /images

*.ai	Original Adobe Illustrator-Dateien
*.fh11	Original Macromedia Freehand-Dateien
*.jpg	Original Rasterbilder
*.eps	Bilder und Grafiken im EPS-Format

Appendix C

Chronologische Liste der Änderungen

2010/11/22 Überarbeitung der originalen Vorlagen von Dr. Wilhelm Burger und Anpassung an die Bedürfnisse einer HTL- Wichtigste Änderungen:

- Wechsel auf UTF8
- Wechsel auf listings2-beta
- Neue Code-Umgebungen für Python und C#
- Vorlagen für Normen und Patente im Literaturverzeichnis
- Hinweise auf DVI-PS Workflow entfernt
- Kapitel zur automatischen LaTeX-Code erzeugung hinzugefügt

Appendix D

LaTeX-Quellcode

Hauptdatei _Diplomarbeit.tex

```
\documentclass[english,oneside,color]{htldipl}
% Zulässige Class Options:
%   Hauptsprache: german (default), english
%   Doppelseitig: oneside (default), twoside
%   Syntax-Highlighting: color (default), black

% die folgende Zeile einkommentieren für Arial-Ähnliche Schriftart
%\renewcommand{\familydefault}{\sfdefault}

\graphicspath{{images/}} % wo liegen die Bilder?
\usepackage[margin=3cm]{geometry} % alle Seitenränder auf 3 cm einstellen

\makeglossaries
%\include{glossary}
\loadglsentries{glossary}
%%-----
\begin{document}
%%-----

\title{GRAMOC - Gradienten Magnetometer Online Controller}
\abteilung{Informatik}
\schwerpunkt{}
\studienort{Wiener Neustadt}
\schule{HTBLuVA Wiener Neustadt}
\schullogo{htl.jpeg}
\abgabejahr{2017/18}
\betreuerA{Dr.\ Michael Stifter}
\betreuerB{}
\betreuerC{}
%\betreuerD{} leer lassen wenn nicht vorhanden
\schuelerA{Nico Kratky}
\evidenzA{5CHIF-13}
\subthemaA{Konstruktion des Versuchstandes}
\schuelerB{Nico LEIDENFROST}
\evidenzB{5CHIF-14}
\subthemaB{Erstellen der Pumpenkennlinien}
\schuelerC{}
\evidenzC{}
\subthemaC{}
\schuelerD{}
\evidenzD{}
\subthemaD{}
```



```

\schuelerE{}
\evidenzE{}
\subthemaE{}
%\schuelerE{} leer lassen wenn nicht vorhanden
%\evidenzE{}
%\subthemaE{}

%%-----
\frontmatter
\maketitle
\tableofcontents
%%-----

\include{vorwort} %ggfs. weglassen
\include{kurzfassung}
\include{abstract}

%%-----
\mainmatter          %Hauptteil (ab hier arab. Seitenzahlen)
%%-----

\include{einleitung}
\include{diplomschrift}
\include{latex}
\include{abbildungen}
\include{mathematik}
\include{literatur}
\include{drucken}
\include{word}
\include{schluss}

%%-----
%%Anhang
\appendix
\printglossaries
\include{anhang_a} % Technische Ergnzungen
\include{anhang_b} % Inhalt der CD-ROM/DVD
\include{anhang_c} % Chronologische Liste der nderungen
\include{anhang_d} % Quelltext dieses Dokuments

%%-----

%Literaturverzeichnis
\clearpage
\addcontentsline{toc}{chapter}{\bibname}

\bibliographystyle{bababrv}
%options: babplain, babunsrt, bababrv, babalpha
%      babplain-fl etc: first/lastname
%      babplain-lf etc: last/firstname
\bibliography{literatur}      %BibTeX-Datei literatur.bib

%%-----

%%Messbox zur Druckkontrolle
\include{messbox}

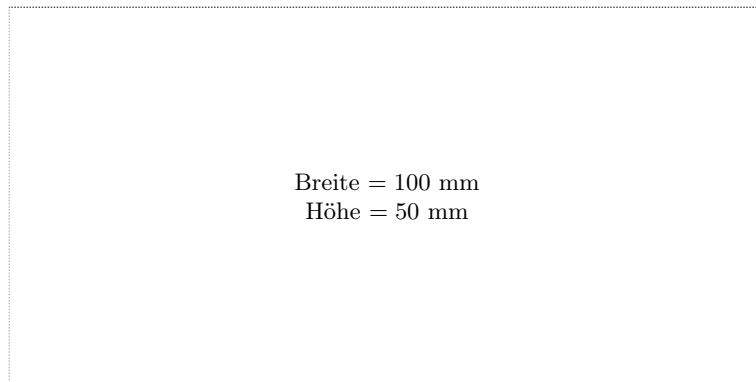
\end{document}

```

Anmerkung: Das sollte nur ein *Beispiel* für die Einbindung von Quellcode in einem Anhang sein. Der LaTeX-Quellcode der eigenen Diplomarbeit ist meist *nicht* interessant genug, um ihn hier wiederzugeben!

Messbox zur Druckkontrolle

— Druckgröße kontrollieren! —



— Diese Seite nach dem Druck entfernen! —