

Konzeption eines Mail Service

ING. THOMAS HERZOG



BACHELORARBEIT

Nr. XXXXXXXXXXXX-B

eingereicht am
Fachhochschul-Bachelorstudiengang

Software Engineering

in Hagenberg

im Februar 2016

Diese Arbeit entstand im Rahmen des Gegenstands

Software Engineering

im

Sommersemester 2016

Betreuer:

Mag. Pjotr I. Czar
Creative Director

Erklärung

Ich erkläre eidesstattlich, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benutzt und die den benutzten Quellen entnommenen Stellen als solche gekennzeichnet habe. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt.

Hagenberg, am 1. Februar 2016

Ing. Thomas Herzog

Inhaltsverzeichnis

Erklärung	iii
Vorwort	viii
Kurzfassung	x
Abstract	xi
1 Einleitung	1
1.1 Zielsetzung	1
1.2 Anforderungen	2
1.3 Überblick Ist-System	3
2 Die Konsolen Applikation	4
2.1 Klassenhierarchie	5
2.2 Implementierung	7
2.2.1 Client-API	9
2.2.2 Datenbank Integration	11
2.2.3 E-Mail Vorlagen	11
2.3 Soll-System	11
2.3.1 Client-API	11
2.3.2 Datenbank Integration	11
2.3.3 E-Mail Vorlagen	11
2.4 Arbeiten in Englisch	11
3 Zum Arbeiten mit LaTeX	12
3.1 Einstieg	13
3.1.1 Software	13
3.1.2 Literatur	14
3.2 Schrift	15
3.2.1 Schriftarten	15
3.2.2 Texte hervorheben	15
3.3 Textstruktur	16
3.3.1 Absatztrennung	16

3.3.2	Überschriften	17
3.3.3	Listen	17
3.3.4	Absatzformatierung und Zeilenabstand	18
3.3.5	Fußnoten	18
3.3.6	Querverweise	19
3.4	Wortabstand und Interpunktion	20
3.4.1	<i>French Spacing</i>	20
3.4.2	Gedanken- und Bindestriche	20
3.4.3	Kommentare	20
3.4.4	Anführungszeichen	21
3.5	Abteilen	22
3.6	Das hagenberg -Paket	22
3.6.1	Einstellungen	23
3.6.2	Definierte Abkürzungen	25
3.6.3	Sprachumschaltung	25
3.6.4	Zusätzliche LaTeX-Pakete	26
3.7	LaTeX-Fehlermeldungen und Warnungen	26
4	Abbildungen, Tabellen, Quellcode	28
4.1	Allgemeines	28
4.2	<i>Let Them Float!</i>	29
4.3	Captions	29
4.4	Abbildungen	30
4.4.1	Wo liegen die Grafikdateien?	31
4.4.2	Grafiken einrahmen	31
4.4.3	Rasterbilder (Pixelgrafiken)	32
4.4.4	Vektorgrafiken	32
4.4.5	TeX-Schriften auch in Grafiken?	34
4.4.6	Abbildungen mit mehreren Elementen	34
4.4.7	Quellenangaben in Captions	35
4.5	Tabellen	36
4.6	Programmtexte	36
4.6.1	Formatierung von Programmcode	37
4.6.2	Platzierung von Programmcode	38
5	Mathem. Formeln etc.	42
5.1	Mathematische Elemente im Fließtext	42
5.2	Freigestellte Ausdrücke	43
5.2.1	Einfache Gleichungen	43
5.2.2	Mehrzeilige Gleichungen	43
5.2.3	Fallunterscheidungen	44
5.2.4	Gleichungen mit Matrizen	44
5.2.5	Verweise auf Gleichungen	45
5.3	Spezielle Symbole	45

5.3.1	Zahlenmengen	45
5.3.2	Operatoren	46
5.3.3	Variable (Symbole) mit mehreren Zeichen	46
5.3.4	Funktionen	46
5.3.5	Maßeinheiten und Währungen	47
5.3.6	Kommas in Dezimalzahlen (Mathematik-Modus)	47
5.3.7	Mathematische Werkzeuge	47
5.4	Algorithmen	48
6	Umgang mit Literatur	50
6.1	Allgemeines	50
6.2	Quellenverweise	51
6.2.1	Das <code>\cite</code> Makro	51
6.2.2	Häufige Fehler	51
6.3	Quellenverzeichnis	52
6.3.1	Literaturdaten in BibTeX	52
6.3.2	Kategorien von Quellenangaben	53
6.3.3	Gedruckte Quellen (<i>literature</i>)	54
6.3.4	Filme und audio-visuelle Medien (<i>avmedia</i>)	60
6.3.5	Online-Quellen (<i>online</i>)	62
6.3.6	Elektronische Datenträger als Ergänzung zur Arbeit .	64
6.3.7	Tipps zur Erstellung von BibTeX-Dateien	64
6.4	Plagiarismus	65
7	Drucken der Diplomarbeit	67
7.1	PDF-Workflow	67
7.2	Drucken	67
7.2.1	Drucker und Papier	67
7.2.2	Druckgröße	67
7.3	Binden	68
7.4	Elektronische Datenträger (CD-R, DVD)	68
8	Schlussbemerkungen	70
8.1	Lesen und lesen lassen	70
8.2	Checkliste	70
A	Technische Informationen	72
A.1	Aktuelle Dateiversionen	72
A.2	Details zur aktuellen Version	72
A.2.1	Allgemeine technische Voraussetzungen	72
A.2.2	Verwendung unter Windows	72
A.2.3	Verwendung unter Mac OS	73
B	Inhalt der CD-ROM/DVD	74

Inhaltsverzeichnis	vii
B.1 PDF-Dateien	74
B.2 LaTeX-Dateien	74
B.3 Style/Class-Dateien	75
B.4 Sonstiges	75
C Chronologische Liste der Änderungen	76
D LaTeX-Quellcode	82
Quellenverzeichnis	84

Vorwort

Folgende Arbeit beschäftigt sich mit der Re-Implementierung einer Mail Lösung, welche verwendet wird um Mails, die innerhalb einer Applikation generiert oder angefordert werden, zu verschicken. Da neue Anforderungen definiert wurden, die sich mit der existierenden Lösung nicht mehr realisieren lassen wurde beschlossen dass diese Lösung reimplementiert werden soll. Hierbei wird diese Arbeit den Fokus auf die existierende wie auch mögliche Implementierung bzw. deren Design legen. Die existierende Lösung weißt ein Alter von 10 Jahren auf, wurde mehrmals erweitert und ist auch dementsprechend designt.

Diese Maillösung wird von der Firma curecomp GmbH verwendet, die eine Cloud basierte SRM-Lösung betreibt (Supplier-RelationShip-Management), deren Prozesse den Versand von E-Mails erfordern wobei diese E-Mails einerseits in verschiedene Bereiche zu unterteilen sind wie z.B.: Systemmails, Lieferverzugsmeldungen, Gutschriften, Bestellbestätigungen usw. und andererseits auch von verschiedenen System der SRM-Lösung erzeugt werden. Bei diesen System handelt es sich einerseits um ein Datenimport/-export System (IIB - IBM-Message-Broker) und andererseits um eine Web Applikation (Websphere - JSF). Beide Systeme sollen in der Lage sein einfach E-Mail Nachrichten eines bestimmten Typs zu versenden, wobei eine Nachricht dynamisch mit Daten versorgt werden soll und auf Basis einer Vorlage die E-Mail erstellt werden soll. Die Anhänge sollen ebenfalls dynamisch von verschiedenen Ressourcen geladen werden können. Eine versendete E-Mail soll über einen bestimmten Zeitraum hinweg weidervesendbar sein und ebenso sollen Log Daten der Metadaten vorhanden sein, die es erlauben den Lebenszyklus einer E-Mail nachträglich nachzuvollziehen. Z.B.: Wann wurde eine E-Mail and wen wie oft von wem versendet.

Beide Systeme sollen mit dem Mail Service System in derselben Art und Weise kommunizieren und interagieren, wobei aber die Systeme eine volle Integration der Client API zur Verfügung gestellt werden soll. Also sollen alle System spezifischen Ressourcen von der Client API nutzbar sein, wodurch ein gewisser Grad der Abstraktion der Implementierung zu gewährleisten ist.

// TODO: Kurzbeschreibung der Systemarchitektur und Komponenten
(halbe Seite) // TODO: Zeichnung der System für die Visualisierung mit
existierenden Mailservice.

Kurzfassung

// TODO: Zusammenfassung nach Abschluss der Arbeit in Deutsch

Abstract

// TODO: Add summary after thesis has been finished in english

Kapitel 1

Einleitung

1.1 Zielsetzung

Diese Bachelorarbeit dient als Vorarbeit für die Re-Implementierung einer existierenden Maillösung der Firma curecomp GmbH, wobei in dieser Arbeit der Fokus auf die Re-Implementierung des Designs gelegt wird. Diese Arbeit wird aufzeigen wie das existierende System in seinen Bereichen wie Client-API, Datenbank Integration und das Erstellen der E-Mails, in weitere Folge 'E-Mail Vorlagen' genannt, zurzeit funktioniert und designt wurde. Dieses existierende Design wird analysiert und ein Verbesserungsvorschlag eingebracht wie ein mögliches neues Design aussehen könnte. Die Konzepte der Implementierung wie die verwendeten Software Muster, die Grundlage der Designentscheidung sowie deren Vorteile und Nachteile werden in dieser Arbeit diskutiert.

Da meiner Meinung nach die existierende Implementierung einige Mängel und Designfehlentscheidungen beinhaltet und dadurch weder flexibel noch erweiterbar ist, wurde der Fokus dieser Arbeit auf das Design dieser Maillösung und dessen Verbesserung gesetzt, wobei diese Arbeit wiederum als Grundlage für die praktische Bachelorarbeit dienen soll, die die Umsetzung der hier entwickelten und diskutierten Konzepte beinhalten soll.

Als Unterstützung wurden folgende beiden literarischen Werke gewählt:

1. Refactoring to patterns¹
2. Refactoring Databases²

¹Author: Joshua Kerievsky, ISBN: 0-321-21335-1

²Author: Scott W.Ambler/Pramod J.Sadalage, ISBN: 0-321-29353-3

1.2 Anforderungen

Über die Zeit haben sich die Anforderungen an das Mailsystem derartig geändert, dass diese nicht mehr mit der existierenden Implementierung umgesetzt werden können. Dies liegt vor allem daran dass die existierende Implementierung aus dem Jahre 2002 stammt und man dessen umgesetztes Design mittlerweile vollständig ausgereizt hat. Andererseits muss man aber auch beachten dass die angeschlossenen Systeme sich über die Zeit evolutioniert, also weiterentwickelt, haben und daher auch neue Anforderungen stellen. Ebenso haben sich auch die Kundenanforderungen geändert. Aufgrund dieser Ausgangssituation wurden folgende technische Anforderungen definiert, die im folgenden aufgelistet sind.

- Persistenz der versendeten E-Mails inklusive Anhänge.
- Integration in andere Systeme in Form von Rest-API (Rest-Service)
- Metadaten sollen für E-Mails definiert werden können (z.B.: Gruppierung)
- Konfigurierbarkeit des Mailversands, Speicherung, Archivierung, ...

Das Gesamtsystem soll hierbei so flexibel wie möglich sein und muss in der Lage sein zumindest mit dem Systemen der Firma curecomp zu interagieren.

1.3 Überblick Ist-System

Der jetzige Mailservice besteht im Grunde nur aus einer Konsolen Applikation, einigen Datenbank Tabellen, und den Implementierungen der DAO (*Data-Access-Object*) der beiden angeschlossenen Systeme. Mann sieht einerseits dass hier die Redundanz in Form der DAO Implementierungen der angeschlossenen Systeme Inkonsistenz mit sich bringt und andererseits dass die Koppelung über die Datenbank unflexibel ist und alle angeschlossenen Systeme mit ihren eigenen Implementierungen darauf zugreifen und es kein übergeordnetes System gibt, welches die allgemeinen Aktionen weg abstrahiert, bündelt und allen Systemen zur Verfügung stellt.

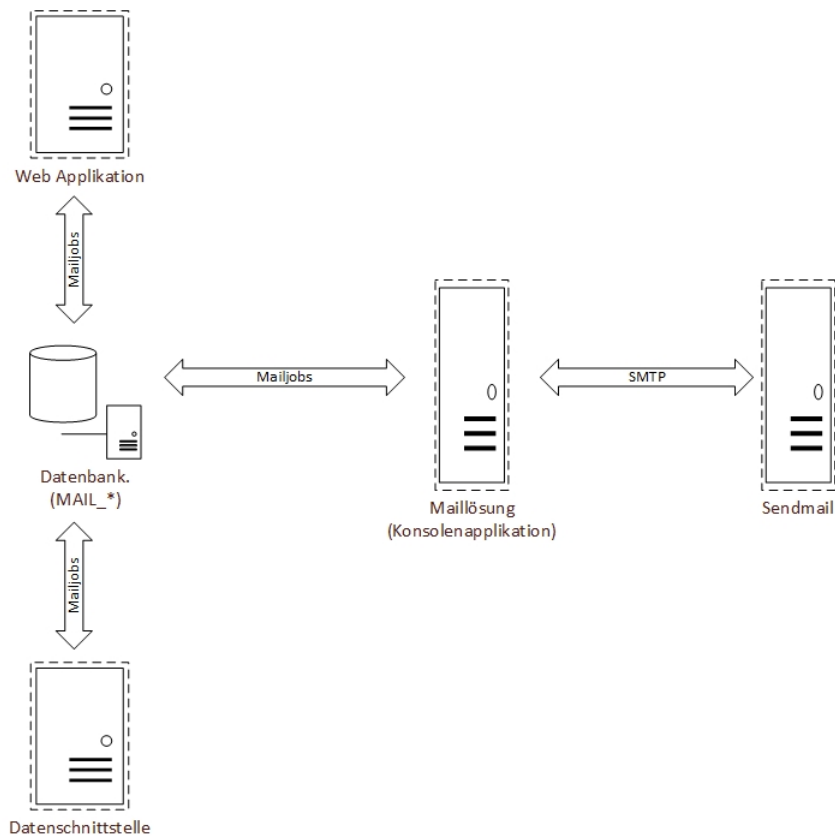


Abbildung 1.1: Diese Abbildung zeigt den konzeptionellen Aufbau der existierenden Maillösung und den jetzigen Kommunikationsweg zwischen den Systemteilnehmern

Kapitel 2

Die Konsolen Applikation

Da in dieser Arbeit das Design der bestehenden Implementierung diskutiert und ein Designvorschlag eingebracht werden soll, muss man zuerst die Struktur und das Design der bestehenden Implementierung verstehen lernen. Da sich die meiste Geschäftslogik in der Konsolen Applikation befindet wird im folgenden dessen Design und Struktur diskutiert.

2.1 Klassenhierarchie

Einleitend betrachten wir die implementierte Klassenhierarchie mit einem Ausschnitt aus der Klassenhierarchie. Dieser Ausschnitt illustriert sehr gut die vorhandene Struktur der Klassenhierarchie.

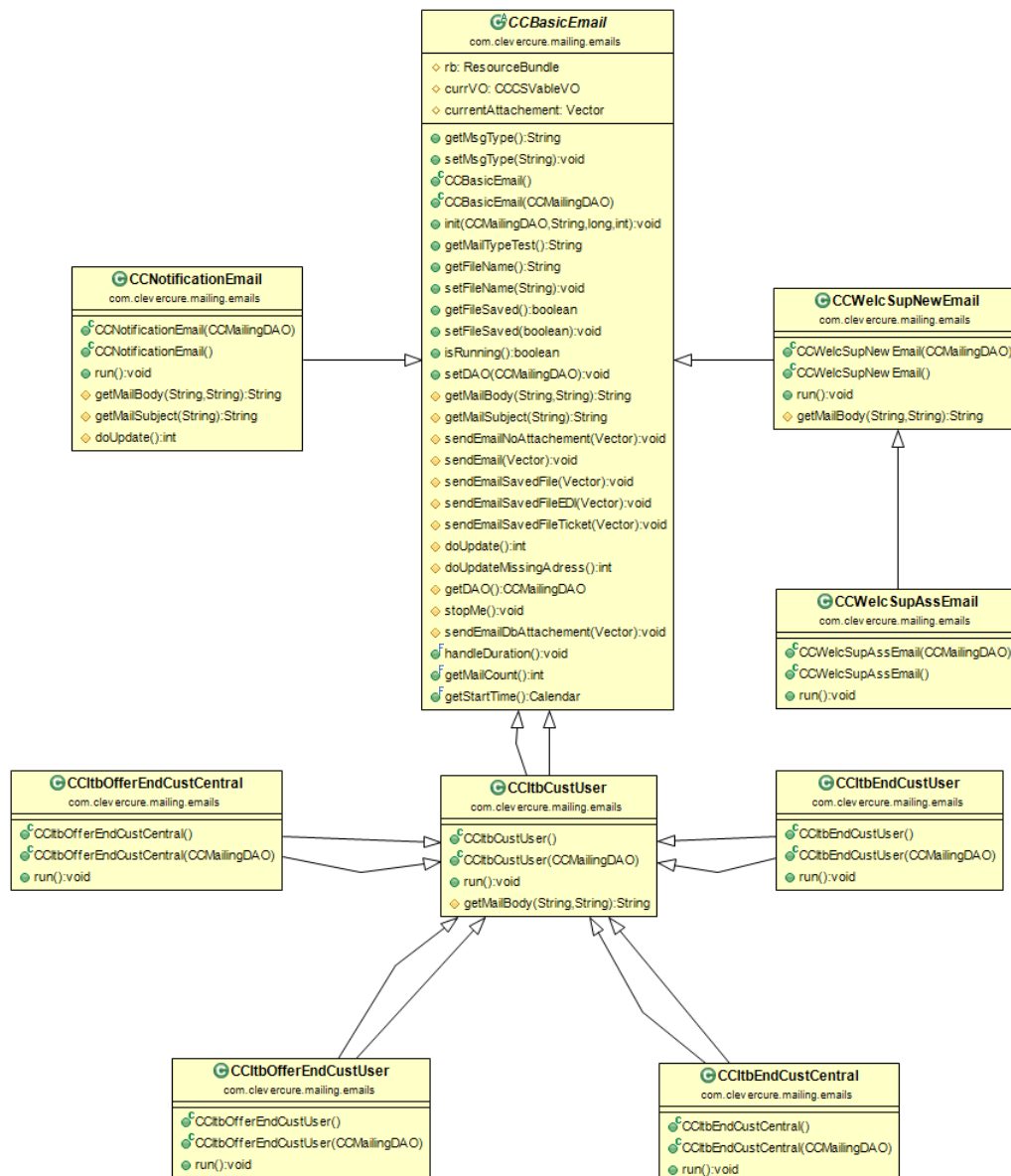


Abbildung 2.1: Die Basisklasse *BasicEmail* mit einigen Implementierungen

Man hat sich hier für das *Factory Method Muster* entschieden wobei die Basisklasse *BasicEmail* eingeführt wurde, welche die gesamte Logik für den

Aufbau und Versand einer E-Mail gekapselt und die einzelnen konkreten Implementierungen in Form von abgeleiteten Klassen, die die einzelnen E-Mail Typen darstellen. Wie im Diagramm bei der Klasse *CCItbCustUser* ersichtlich wurden auch E-Mail Typen implementiert, die ihrerseits wieder als Basisklasse dienen für untergeordnete E-Mail Typen. Aufgrund des vergebenen Namens kann man annehmen dass es sich hier um E-Mail Benachrichtigungen für einen Benutzer eines Kunden des Moduls ITB handelt, die man versucht hat zu gruppieren.

Die Gruppierung wurde hier durch Einführung einer eigenen Vererbungshierarchie auf Basis einer eingeführten Basisklasse *CCItbCustUser*, die ihrerseits die eigentliche Basisklasse *BasicEmail* weg abstrahiert, realisiert.

2.2 Implementierung

Da wir nun die Klassenhierarchie kennen wenden wir uns deren Implementierung zu. Im folgenden werden die Implementierungen der Klasse *CCItbCustUser* und dessen Ableitungen diskutiert. Im Punkt ?? wurde behauptet dass diese Ableitungen eingeführt wurden um E-Mail Typen zu gruppieren. Man könnte aber auch davon ausgehen dass diese eigene Vererbungshierarchie eingeführt wurde um Funktionalitäten für die einzelnen E-Mails zu kapseln. Analysieren wir nun diese Implementierungen um zu sehen welche Funktionalitäten in einen E-Mail Typ implementiert wurden.

```
// TODO: Add source of 'CCItbCustUser' and its decendants.  
// TODO: CCItbCustUser
```

Im folgenden wird das existierende Design der Implementierung in den drei Bereichen

1. *Client-API*

Wie versenden bzw. generieren die Systeme die E-Mails

2. *Datenbank Integration*

Wie ist das Datenbankdesign und welche Daten werden gespeichert und warum

3. *E-Mail Vorlagen*

Wie werden die Inhalte der E-Mail generiert bzw. erstellt.

Da der Mailversand in Java verhältnismäßig einfach ist wird auf diesen Teil nicht genauer eingegangen sehr wohl aber die Funktionalität im weiteren Verlauf dieser Arbeit diskutiert. Die oben genannten drei Unterteilungen stellen die Hauptteile der Maillösung dar und werden auch in einem neuen Design so benötigt ohne Rücksichtnahme auf die konkrete Art der Implementierung.

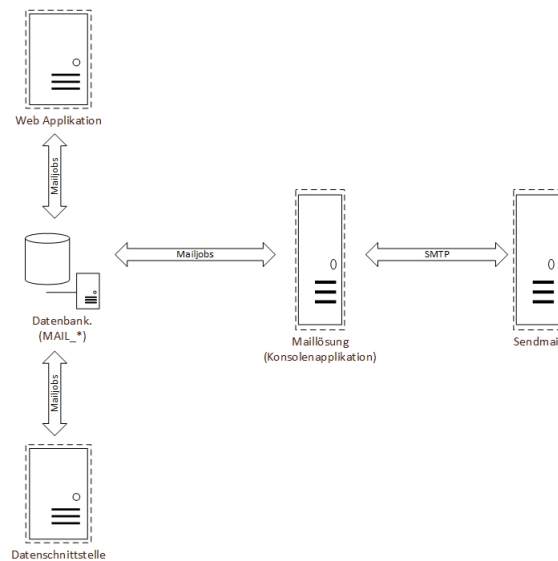


Abbildung 2.2: Diese Abbildung zeigt den konzeptionellen Aufbau der existierenden Maillösung und den jetzigen Kommunikationsweg zwischen den Systemteilnehmern

In dieser Abbildung können wir sehen dass es nur einen Kommunikationsweg zwischen den Systemteilnehmer gibt und zwar die Kommunikation über die Datenbank. Dies bedeutet es werden sogenannte Mailjobs erstellt, die einen Eintrag in einer Tabelle der Datenbank darstellen, welche alle Daten für die E-Mail zur Verfügung stellen und von der Maillösung ausgelesen und verarbeitet werden.

Hierbei enthält die Maillösung alle Vorlagen für alle implementierten E-Mails und SQL Abfragen, die die Daten aus der Datenbank auslesen und diese ausgelesenen Daten als Parameter and das Vorlagensystem weitergibt damit dieses die tatsächliche E-Mail Nachricht erstellen kann. Hierbei ist die Aktualität der Daten kritisch, da die E-Mail zeitversetzt zum erstellten Mailjob generiert wird und sich dadurch auch die Daten, die aus der Datenbank ausgelesen werden, sich zwischenzeitlich ändern könnten. Ebenso müssen die Daten separat und meistens erneut aus der Datenbank ausgelesen werden, obwohl in den meisten Fällen die Daten beim Erstellen des Mailjobs bereits vorhanden und vor allem zu diesem Zeitpunkt auch aktuell waren. Es gibt zwar Fälle in denen zeitgesteuert E-Mails generiert werden, wie z.B.: einmal täglich Lieferverzugsmeldungen, in denen der aktuelle Status der Datenbank eine Rolle spielt. Dieser Typ der E-Mails stellt jedoch im Verhältnis zu den *'sofort versendeten'* E-Mails nur einen kleinen Prozentanteil dar und hier ist das Problem der Aktualität der verwendeten Daten nicht vorhanden. Bezüglich der Aktualität der Daten sei noch angemerkt dass das erneute Versenden einer E-Mail im bestehenden System nicht mit konsistent Daten möglich ist, da die Daten jedes mal erneut aus der Datenbank ausgelesen werden und der ursprüngliche Zustand der Daten nicht garantiert zur Verfügung steht, da diese sich zwischenzeitlich hätten ändern können, was auch nicht nachvollziehbar wäre.

2.2.1 Client-API

Da die Kommunikation ausschließlich über die Datenbank erfolgt ist eine echte Client API nicht vorhanden. Alle Systeme implementieren das Erstellen der Mailjob Einträge selbstständig und es wird keine gemeinsame Client API verwendet, was einen gewissen Grad der Inkonsistenz bei Änderungen der Datenstruktur sowie der Semantik der Daten mit sich bringt. Als Beispiel wird hier die Semantik der einzelnen Spalten angeführt, die zwar technisch sich innerhalb einer Datentyp Domain befinden, sich die semantische Bedeutung der enthaltenen Daten aber unterscheidet. Dies ist zwar Teil der Datenbank Integration aber die Tatsache dass diese Semantik rein innerhalb der Applikation abgebildet werden kann und nicht über die Datenbankfunktionalitäten wie z.B.: Fremdschlüssel wird dieser Teil hier diskutiert. Da die Maillösung über SQL Abfragen die Daten für die E-Mail Nachricht erhält müssen auch Parameter bereitgestellt werden, die in der SQL Abfrage gesetzt werden. Diese Parameter wurden *'generisch'* über eine

festgesetzte Anzahl von Tabellenspalten abgebildet, dessen enthaltene Daten je nach E-Mailtyp anders interpretiert werden. Es kann also sein das ein Eintrag einer Tabellen für die Spalte *COL_1* einen String enthält mit dem Wert *'Thomas Herzog'* und ein anderer Eintrag einen string mit dem Wert *'14'* der in der SQL Abfrage aber als Integer Datentyp behandelt wird. Die Systeme die Mailjobs erzeugen müssen sich also dem Kontext einer E-Mail bewusst sein, sowie müssen berücksichtigen, dass die richtigen Spalten der Tabelle *MAIL_JOB* mit den richtigen Daten befüllt werden, wobei auch beachtet werden muss in welchen Datentyp der Datensatz in weiterer Folge durch die Maillösung interpretiert wird.

Dies ist in meinen Augen sehr inkonsistent und fehleranfällig und hat auch schon einige Probleme verursacht.

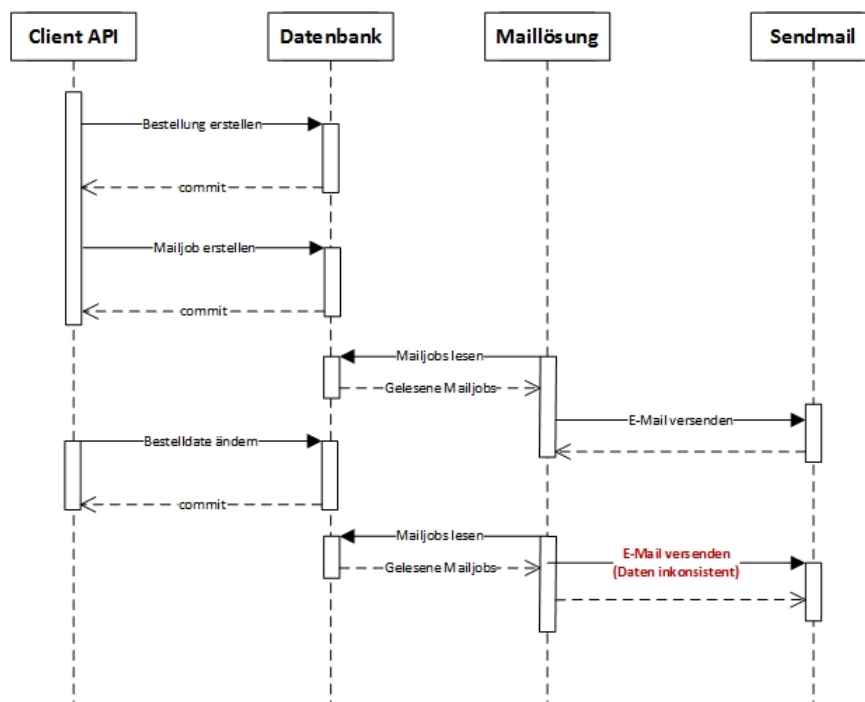


Abbildung 2.3: Diese Abbildung zeigt das Problem der inkonsistenten Daten beim erneuten E-Mailversand einer bereits versendeten E-Mail, mit dem Beispiel einer angelegten und anschließend geänderten Bestellung, auf

2.2.2 Datenbank Integration

2.2.3 E-Mail Vorlagen

2.3 Soll-System

2.3.1 Client-API

2.3.2 Datenbank Integration

2.3.3 E-Mail Vorlagen

2.4 Arbeiten in Englisch

Kapitel 3

Zum Arbeiten mit LaTeX

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 für den Anfänger 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 äußerste 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. Allerdings sollte mithilfe von LaTeX ein professionell aussehendes Ergebnis einfacher zu erreichen sein und man wird sich wohl auch einigen Ärger mit Fehlern und Einschränkungen gängiger Software ersparen. Zudem könnte es durchaus sein, dass man nebenbei auch sein Auge für die Feinheiten des Buchsatzes (weiter-)entwickelt.²

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

²Dieses abschließende Textelement wurde übrigens zur Ermöglichung eines Zeilenumbruchs nach der Klammer so gesetzt: `...(\weiter-){\optbreaknh}entwickelt`. Das Makro `\optbreaknh` („optional break with no hyphen“) ist in `hgb.sty` definiert.

3.1 Einstieg

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 \LaTeX 2_ε (sprich „LaTeX zwei e“). Zum Arbeiten mit LaTeX benötigt man zwei Dinge:

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

Sämtliche Komponenten sind kostenlos und für alle gängigen Plattformen verfügbar.

Windows

Unter *Windows* (XP, Win7) hat sich folgendes Setup bewährt, mit dem u. a. auch dieses Dokument erstellt wurde:

- **LaTeX-Distribution:** *MikTeX* 2.9 oder höher. MikTeX³ enthält auch alle notwendigen Hilfsprogramme, wie beispielsweise **bibtex**.
- **PDF-Viewer:** *SumatraPDF*⁴
- **Frontend:** *TeXnicCenter*.⁵ Grundsätzlich kann man jeden Text-Editor⁶ verwenden, praktischer ist jedoch eine integrierte LaTeX-Umgebung wie TeXnicCenter, die einen auch bei der Dateiverwaltung, der Verarbeitung der Dokumente und der Fehlerbehandlung unterstützt. Eine interessante Alternative ist die Verwendung von *Eclipse*⁷ als plattformunabhängiges Frontend (mit dem *TeXlipse*⁸ Plugin).

Beim ersten Mal sollten *MikTeX*, *SumatraPDF* und *TeXnicCenter* in genau dieser Reihenfolge installiert werden.

Mac OS

Unter Mac OS X ist die Referenzdistribution *MacTeX*.⁹ Sie enthält neben der TeX-Distribution *TeX Live* auch gängige Editoren wie *TeXWorks* oder

³www.miktex.org

⁴<http://blog.kowalczyk.info/software/sumatrapdf/>

⁵www.texniccenter.org

⁶Unter Windows z. B. *UltraEdit* (www.ultraedit.com) oder *WinEdit* (www.winedit.com).

⁷www.eclipse.org

⁸<http://texlipse.sourceforge.net/>

⁹<http://www.tug.org/mactex>

TeXshop. Mit dem *TeX Live Utility* können Pakete verwaltet und die Distribution auf den neuesten Stand gebracht werden. Als Alternative zu den beiden genannten Editoren steht *TeXnicle*¹⁰ zur Verfügung. Er bietet – ähnlich wie *TeXnicCenter* unter Windows – einen projektbasierten Workflow an. Ein PDF-Viewer muss unter Mac OS X übrigens nicht extra installiert werden. Alle genannten Editoren beinhalten eine eigene PDF-Vorschau.

Linux

Auch unter Linux ist *TeX Live* (s. oben) eine häufig verwendete TeX-Distribution. Als Frontend sind beispielsweise *Lyx*¹¹, *Kile*¹² und *Texmaker*¹³ verbreitet. In manchen gängigen Linux-Versionen ist bereits eine komplette LaTeX-Distribution enthalten, sodass im besten Fall überhaupt keine zusätzliche Installation notwendig ist.

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. [**Schmidt01**]
- *The Not So Short Introduction to L^AT_EX 2_ε* von Oetiker et al. [**Oetiker01**]

Als mittlerweile bereits klassisches Handbuch zu LaTeX ist

- *A Guide to L^AT_EX 2_ε* von H. Kopka und P. Daly [**Kopka98**]

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

www.ctan.org¹⁴
<http://dante.ctan.org>¹⁵
www.tex.ac.uk

Besonders nützlich sind auch die *Comprehensive List of LaTeX Symbols* [**Pakin01**] und die Beschreibungen wichtiger LaTeX-Pakete, wie

`babel` [**Braams2008**],
`graphics`, `graphicx` [**Carlisle99**],

¹⁰<http://www.bobsoft-mac.de/texnicle/texnicle.html>

¹¹www.lyx.org

¹²<http://kile.sourceforge.net>

¹³www.xmlmath.net/texmaker/

¹⁴www.ctan.org/tex-archive/help/Catalogue/bytopic.html

¹⁵<http://dante.ctan.org/tex-archive/help/Catalogue/bytopic.html>

```
fancyhdr [Oostrum97],
caption [Sommerfeldt07],
subfig [Cochran05].
```

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 Knuth¹⁶ entwickelt wurden. Die drei Basis-Schrifttypen 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.

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>

¹⁶www-cs-staff.stanford.edu/~knuth/

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

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

- *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 Satz eigentlich überflüssig. Sie sollten daher nur in Ausnahmefällen verwendet werden, z. B.

`\underline{überflüssig}`¹⁷

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.¹⁸ 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 bewährt¹⁹ und sollte auch beibehalten werden, es sei denn man hat wirklich *sehr* gute Gründe dagegen.

¹⁷Unterstrichene Texte werden zudem nicht automatisch abgeteilt.

¹⁸Das ist die Standardeinstellung in LaTeX und natürlich abhängig von der verwendeten Dokumentenklasse, Style etc.

¹⁹Wer es nicht glaubt, sollte sein Bücherregal (oder notfalls das seiner Eltern) nach Gegenbeispielen durchsuchen.

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.²⁰

3.3.2 Überschriften

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

```
\part{Titel}21
\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 Interpunktion (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 („Knödeliste“), geordnete Auflistung (Aufzählung) und Beschreibungsliste (Description):

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

²⁰Das Makro `\SuperPar` ist in `hgb.sty` definiert.

²¹`\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.

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. 4.2 auf Seite 37).

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.

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 8, die folgendermaßen definiert ist:

```
\chapter[Schlussbemerkungen]%
{Schlussbemerkungen%
\protect\footnote{Diese Anmerkung ...}}%
```

Dabei ist der erste (optionale) Titel `[Schlussbemerkungen]` 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 hier übrigens notwendig, um eventuelle Leerzeichen, die durch Zeilenumbrüche im Quelltext entstehen, zu eliminieren (diesen Trick braucht man in LaTeX häufig, s. Abschn. 3.4.3). 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.²²

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} ..
```

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
```

²²Das 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.

3.4 Wortabstand und Interpunktion

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 dem ~ Zeichen 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 Gedankenstrichen²³ gibt es eindeutige Konventionen:

1. Im *Deutschen* setzt man üblicherweise einen von zwei Leerzeichen umgebenen Gedankenstrich²⁴ – 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 Gedankenstrich²⁵ 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:

²³Für alle drei gibt es übrigens auch in *Word* entsprechende Sonderzeichen.

²⁴Halbgeviertstrich (*En Dash*).

²⁵Geviertstrich (*Em Dash*).

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 Zeilenende das Entstehen eines Leerzeichens vor einer nachfolgenden Fußnotenmarke vermeiden kann:

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

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

```
... normaler Text.
%
\begin{center}
    Dieser Test 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 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).²⁶

3.6 Das hagenberg-Paket

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

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

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

²⁶ Angeblich waren eigenständige Textänderungen durch Schriftsetzer auch beim früheren Bleisatz durchaus üblich.

3.6.1 Einstellungen

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

```
\documentclass[type,language]{hgbthesis}
```

Dabei sind die möglichen Optionen für *type*

`master` (Masterarbeit = *default*)

`diplom` (Diplomarbeit)

`bachelor` (Bachelorarbeit)

`praktikum` (Praktikumsbericht)

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

`german` (*default*)

`english`

Wird überhaupt keine Option angegeben, dann wird als Default-Einstellung [`diplom`, `german`] verwendet. Der vollständige Quelltext für eine entsprechende `.tex` Hauptdatei ist in Anhang D gelistet.

Angaben zur Arbeit

Die Dokumentenklasse ist für verschiedene Arten von Arbeiten vorgesehen, die sich nur im Aufbau der Titelseiten unterscheiden. Abhängig vom gewählten Dokumententyp sind unterschiedliche Elemente für die Titelseiten erforderlich (siehe Tabelle 3.1). Folgende Basisangaben sind für **alle** Arten von Arbeiten erforderlich:

```
\title{Titel der Arbeit}
\author{Autor}
\studiengang{Studiengang}
\studienort{Studienort}
\abgabedatum{yyyy}{mm}{dd}
```

Für **Bachelorarbeiten** werden zusätzlich zu den Basisangaben folgende Elemente verwendet (bei Diplom- und Masterarbeiten nicht relevant):

```
\nummer{laufende Nummer der Arbeit}27
\gegenstand{Gegenstand oder Projektlehrveranstaltung}
\semester{Semester der Lehrveranstaltung}
\betreuer{Name} oder \betreuerin{..}
```

Für **Praktikumsberichte** werden zusätzlich zu den Basisangaben folgende Elemente berücksichtigt:

²⁷Wird normalerweise von der Institution vergeben. An der FH-Hagenberg ist dies bei einer Bachelorarbeit die (10-stellige) Studenten-ID des Autors, z. B. 0310238045-A für die erste Bachelorarbeit.

Tabelle 3.1: Elemente in Titelseiten für verschiedene Dokumentenoptionen.

<i>Element</i>	<i>diplom</i>	<i>master</i>	<i>bachelor</i>	<i>praktikum</i>
<code>\title</code>	+	+	+	+
<code>\author</code>	+	+	+	+
<code>\studiengang</code>	+	+	+	+
<code>\studienort</code>	+	+	+	+
<code>\abgabedatum</code>	+	+	+	+
<code>\nummer</code>	–	–	+	+
<code>\gegenstand</code>	–	–	+	–
<code>\betreuer</code>	–	–	+	+
<code>\firma</code>	–	–	–	+
<code>\firmenTel</code>	–	–	–	+
<code>\firmenUrl</code>	–	–	–	+

`\nummer{laufende Nummer der Arbeit}`
`\betreuer{Betreuer im Unternehmen} oder \betreuerin{..}`
`\firma{name und Adresse der Firma}`
`\firmenTel{Telefonnummer der Firma}`
`\firmenUrl{Web-Site der Firma}`

Titelseiten

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

`\maketitle`

automatisch generiert, abhängig vom Typ der Arbeit und den obigen Einstellungen:

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

Auf der Copyright-Seite werden auch die Bedingungen für die Nutzung und Weitergabe der Arbeit vermerkt. Der zugehörige Text kann durch folgenden Einstellungen am Beginn des Dokuments bestimmt werden:

`\cclicense`

Veröffentlichung unter einer Creative Commons²⁸ Lizenz, die die freie Weitergabe der Arbeit unter Nennung des Autors, jedoch keine kommerzielle Nutzung oder Bearbeitung erlaubt (Standardeinstellung).

`\strictlicense`

Traditionelle Einschränkung der Nutzungsrechte (*Alle Rechte vorbehalten* bzw. *All Rights Reserved*).

²⁸<http://creativecommons.org/licenses/by-nc-nd/3.0/>

Tabelle 3.2: In `hgb.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	<code>\etc</code>	etc.

`\license{Lizenztext}`

Damit kann alternativ ein eigener *Lizenztext* angegeben werden, falls notwendig. Solche Änderungen sollte man natürlich unbedingt mit seiner Hochschule abstimmen.

3.6.2 Definierte Abkürzungen

Es wird im **hagenberg**-Paket weiters eine Reihe von Abkürzungsmakros²⁹ definiert, die das Schreiben vereinfachen und für konsistente Zwischenabstände sorgen (Tab. 3.2). Bei der Verwendung von Makros ist allgemein zu beachten, dass sie nachfolgende Leerzeichen manchmal „auffressen“, so dass vor dem nachfolgenden Text kein Abstand erzeugt wird.³⁰ Dagegen kann man sich notfalls mit einem nachfolgenden „\ “ oder umhüllenden `{}`-Klammern behelfen. Bei Verwendung von Makros mit abschließendem Punkt an einem Satzende sollte man auch darauf achten, dass keine *doppelten* Punkte gesetzt werden.

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 **hagenberg**-Paket zwei spezielle *Environments* vorgesehen:

```
\begin{english}
This is a 1-page (maximum) summary
```

²⁹In Anlehnung an den `jkthesis`-Style von Jochen Küpper (www.jochen-kuepper.de).

³⁰Bei fast allen in `hgb.sty` definierten Makros wird dies allerdings durch den Einsatz von `\xspace` verhindert.

```

        of your work in English.
    \end{english}
    \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 „`ngerman`“ (*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. 3.3). Diese Pakete werden am Anfang durch das `hagenberg`-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. 3.1.2 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/cmr/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

Tabelle 3.3: Im `hagenberg`-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>	Mathematischer Satzsatz
<code>amsfonts, amsbsy</code>	Mathematische Symbole
<code>babel</code>	Sprachumschaltung
<code>biblatex</code>	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>listings</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

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), ...
```

Kapitel 4

Abbildungen, Tabellen, Quellcode

4.1 Allgemeines

Abbildungen (*figures*) und Tabellen (*tables*) werden üblicherweise zusammen mit einem nummerierten Titel (*caption*) zentriert angeordnet (siehe Abb. 4.1). Im Text *muss* es zu jeder Abbildung einen Verweis geben und die eigentliche Abbildung sollte erst *nach* dem ersten Verweis platziert werden.



Abbildung 4.1: Coca-Cola Werbung 1940 [CocaCola1940].

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 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 zurecht kommt, sollte man (durch gezielte Platzierungsanweisungen [Oetiker01]) 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=.95\textwidth]{cola-public-domain-photo-p}
4 \caption{Coca-Cola Werbung 1940 \cite{CocaCola1940}.}
5 \label{fig:CocaCola}
6 \end{figure}
```

Der Name des Labels (`fig:CocaCola`) kann beliebig gewählt werden. Die Kennzeichnung `fig:` ist (wie in Abschn. 3.3.6 erwähnt) nur eine nützliche

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.

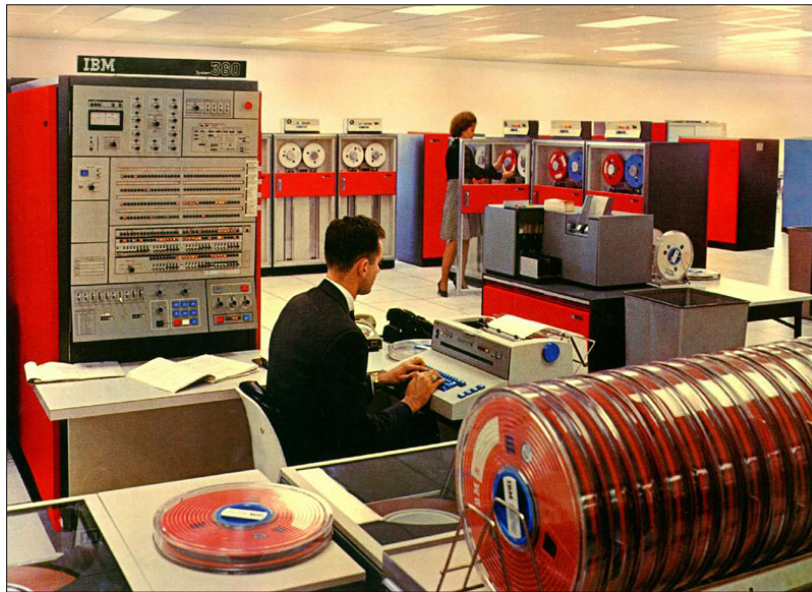


Abbildung 4.2: Beispiel für einen langen Caption-Text. UNIVAC brachte 1961 mit dem Modell 751 den ersten Hochleistungsrechner mit Halbleiterspeicher auf den Markt. Von diesem Computer wurden in den U.S.A. bereits im ersten Produktionsjahr über fünfzig Exemplare verkauft, vorwiegend an militärische Dienststellen, Versicherungen und Großbanken. Die Ablöse erfolgte zwei Jahre später durch das zusammen mit SPERRY entwickelte Modell 820. Das klingt vielleicht plausibel, ist aber völliger Unsinn, denn das Bild zeigt in Wirklichkeit eine System/360 Anlage von IBM. Bildquelle [IBM360].

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. 4.1) oder eine längere (Abb. 4.2). 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` [Carlisle99] empfohlen (wird durch das `hagenberg`-Paket bereits eingebunden). Mit dem aktuell verwendeten Workflow (`pdflatex`)

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.

können Bild- bzw. Grafikformate ausschließlich in folgenden Formaten eingebunden werden:

- **PNG**: für Grau-, S/W- und Farb-Rasterbilder (bevorzugt),
- **JPEG**: für Fotos (wenn nicht anders vorhanden),
- **PDF**: für Vektorgrafiken (Illustrationen, Strichzeichnungen etc.).

Bei Rasterbildern sollte wenn möglich PNG verwendet werden, weil die darin enthaltenen Bilder verlustfrei komprimiert sind und daher keine sichtbaren Kompressionsartefakte aufweisen. Im Gegensatz dazu sollte man JPEG nur dann verwenden, wenn das Originalmaterial (Foto) bereits in dieser Form vorliegt.

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 `_DaBa.tex` (s. auch Anhang D):

```
\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]{ibm-360-color}
\includegraphics[scale=1.5]{ibm-360-color}
```

Man beachte, dass dabei die Dateierweiterung 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 `hgb.sty`) kann man optional einen dünnen Rahmen rund um die Grafik erzeugen, z. B.:

```
\FramePic{\includegraphics[height=50mm]{ibm-360-color}}
```

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

In der Regel sollte man Bilder, die für den Einsatz in Druckdokumenten gedacht sind, nicht mit verlustbehafteten Kompressionsverfahren abspeichern. Insbesondere sollte man die Verwendung von JPEG möglichst vermeiden, auch wenn viele Dateien dadurch wesentlich kleiner werden. 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. 4.3).

4.4.4 Vektorgrafiken

Für schematische Abbildungen (z. B. Flussdiagramme, Entity-Relationship-Diagramme oder sonstige strukturelle Darstellungen) sollte man unbedingt Vektorgrafiken (PDF) 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* oder *Illustrator* von *Adobe*. Manche gängige

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

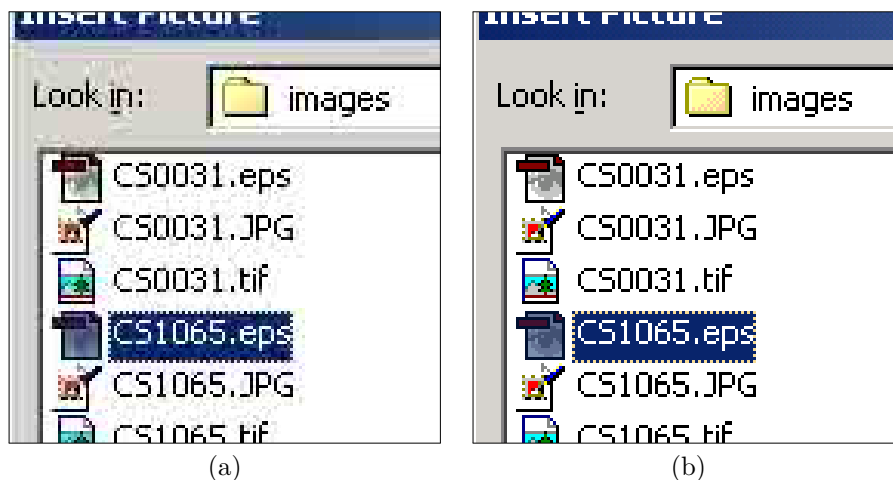


Abbildung 4.3: Typischer JPEG-Pfusch. Screenshots und ähnliche im Original verfügbare Rasterbilder sollten für Druckdokumente *keinesfalls* mit JPEG komprimiert werden. Das Ergebnis (a) sieht gegenüber dem unkomprimierten Original (b) nicht nur schmutzig aus, sondern wird im Druck auch schnell unleserlich.

Grafikprogramme unterstützen allerdings keinen direkten Export von PDF-Dateien oder erzeugen unsaubere Dateien. Vor der Entscheidung für eine bestimmte Zeichensoftware sollte man das im Zweifelsfall ausprobieren. PDF kann im Notfall über einen entsprechenden Druckertreiber erzeugt werden.

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:** Hairli-

nes vermeiden und stattdessen immer konkrete Strichstärken (≥ 0.25 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.

***BaKoMa*-Schriften (TrueType)**

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 z. B. 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.

***Latin Modern Roman* Fonts (OpenType)**

Eine Alternative dazu sind die „LM-Roman“⁶ Open-Type Schriften, die speziell für die Verwendung im Umfeld von LaTeX entwickelt wurden. Sie sind auch Teil der MikTeX-Installation.⁷ Diese Schriften enthalten u. a. Zeichen mit Umlauten und sind daher auch für deutsche Texte recht bequem zu verwenden.

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. 4.4 dargestellt. Im Text könnte ein Verweis auf einen einzelnen Teil der Abbildung, etwa das einreihige Rollenlager in Abb. 4.4 (c), so aussehen:

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

Für kompliziertere Abbildungen sollte man die Verwendung des `subfig`-Pakets [Cochran05] in Betracht ziehen.

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/.

⁶<http://www.gust.org.pl/projects/e-foundry/latin-modern>

⁷C:/Program Files (x86)/MikTeX 2.9/fonts/opentype/public/lm/

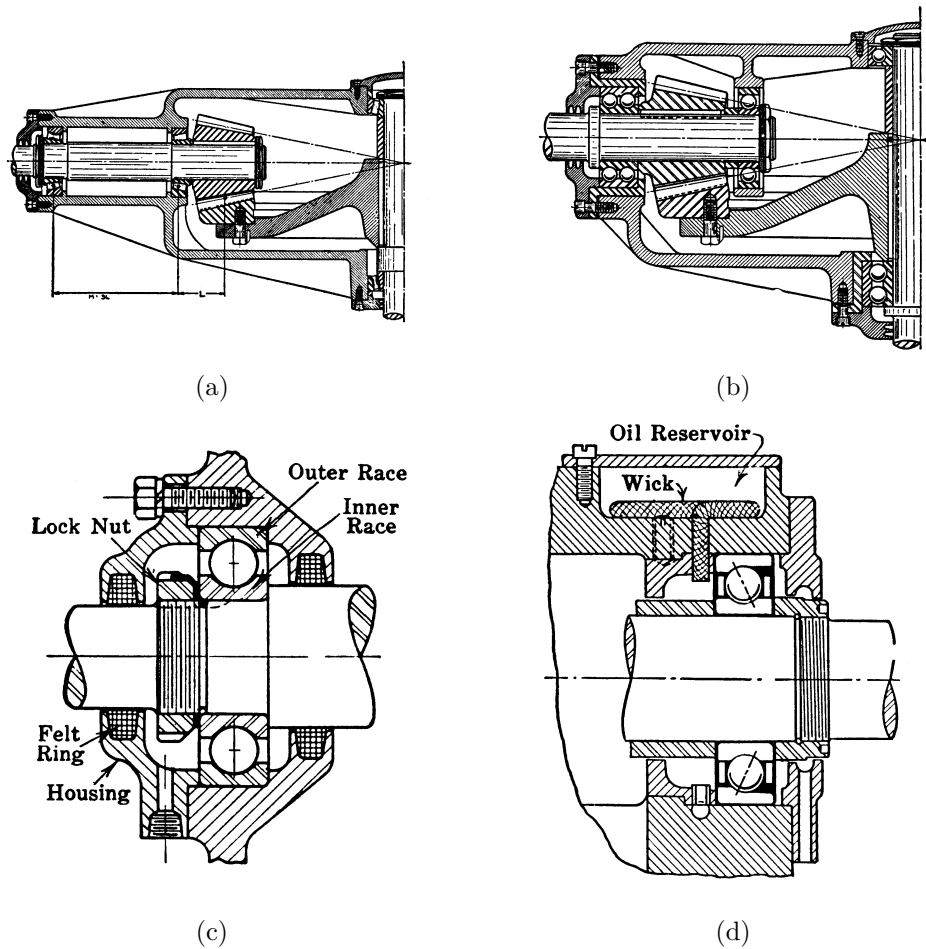


Abbildung 4.4: Diverse Maschinenelemente als Beispiel für eine Abbildung mit mehreren Elementen. Overhang Mounting (a), Straddle Mounting (b), einreihiges Rollenlager (c), Schmierung von Rollenlagern (d). Die Abbildung verwendet im oberen Teile eine 2×2 Tabelle (`tabular`), in der die Breite der Spaltenränder gesondert spezifiziert ist (Details finden sich im Quelltext). Bildquelle [Faires34].

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. 4.4 demonstriert. Weitere Details zu dieser Art von Quellenangaben finden sich in Kap. 6 (insbes. Abschnitt 6.3.5).

Tabelle 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

Programm 4.1: LaTeX Quelltext zu Tab. 4.1. Die Erzeugung des dargestellten Listings selbst ist in Abschn. 4.6 beschrieben.

```

\begin{table}
\caption{Prozessor-Familien im Überblick.}
\label{tab:processors}
\centering
\setlength{\tabcolsep}{5mm} % separator between columns
\def\arraystretch{1.25} % vertical stretch factor
\begin{tabular}{|r||c|c|c|}
\hline
& \emph{PowerPC} & \emph{Pentium} & \emph{Athlon} \\
\hline
Manufacturer & Motorola & Intel & AMD \\
\hline
Speed & high & medium & high \\
\hline
Price & high & high & medium \\
\hline
\end{tabular}
\end{table}

```

4.5 Tabellen

Tabellen werden häufig eingesetzt um numerische Zusammenhänge, Testergebnisse etc. in übersichtlicher Form darzustellen. Ein einfaches Beispiel ist Tab. 4.1, der LaTeX-Quelltext dazu findet sich in Prog. 4.1.

Manchmal ist es notwendig, in Tabellen relativ viel Text in engen Spalten unter zu bringen, wie in Tab. 4.2. In diesem Fall ist es sinnvoll, auf den Blocksatz zu verzichten und 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.

Tabelle 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

Tabelle 4.3: In `hgb.sty` vordefinierte Code-Umgebungen.

C (ANSI):	<code>\begin{CCode}</code>	<code>... \end{CCode}</code>
C++ (ISO):	<code>\begin{CppCode}</code>	<code>... \end{CppCode}</code>
Java:	<code>\begin{JavaCode}</code>	<code>... \end{JavaCode}</code>
JavaScript:	<code>\begin{JsCode}</code>	<code>... \end{JsCode}</code>
PHP:	<code>\begin{PhpCode}</code>	<code>... \end{PhpCode}</code>
HTML:	<code>\begin{HtmlCode}</code>	<code>... \end{HtmlCode}</code>
CSS:	<code>\begin{CssCode}</code>	<code>... \end{CssCode}</code>
XML:	<code>\begin{XmlCode}</code>	<code>... \end{XmlCode}</code>
LaTeX:	<code>\begin{LaTeXCode}</code>	<code>... \end{LaTeXCode}</code>
Generisch:	<code>\begin{GenericCode}</code>	<code>... \end{GenericCode}</code>

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, insbesondere das `listings`-Package.⁸ Damit sind auch die in Tabelle 4.3 aufgelisteten Code-Umgebungen realisiert. Die Verwendung ist äußerst einfach, z. B. für Quellcode in der Programmiersprache C schreibt man

```
\begin{CCode}
...
\end{CCode}
```

Der Quellcode innerhalb dieser Umgebungen wird in der jeweiligen Programmiersprache interpretiert, wobei Kommentare erhalten bleiben. Diese

⁸<http://www.ctan.org/tex-archive/macros/latex/contrib/listings/>

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. mit

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

Ein Beispiel mit Java ist in Prog. 4.2 gezeigt, wobei der oben angeführte Label in Zeile 14 steht. Man beachte, dass innerhalb der Kommentare auch mathematischer Text (wie etwa in Zeile 21 von Prog. 4.2) stehen kann.

Nummerierung der Code-Zeilen

Alle in Tabelle 4.3 angeführten Code-Umgebungen können mit optionalen Argumenten verwendet werden, die insbesondere zur Steuerung der Zeilennummerierung hilfreich. Im Normalfall (also ohne zusätzliche Angabe) mit

```
\begin{someCode} ...
```

werden alle Code-Zeilen (einschließlich der Leerzeilen) bei 1 beginnend und fortlaufend nummeriert. Bei aufeinanderfolgenden Codesegmenten ist es oft hilfreich, die Nummerierung aus dem vorherigen Abschnitt kontinuierlich weiter laufen zu lassen, ermöglicht durch die Angabe des optionalen Arguments `firstnumber=last`:

```
\begin{someCode}[firstnumber=last] ...
```

Um die Nummerierung der Codezeilen gänzlich zu unterbinden genügt die Angabe des optionalen Arguments `numbers=none`:

```
\begin{someCode}[numbers=none] ...
```

In diesem Fall ist natürlich die Verwendung von Zeilenlabels im Code nicht sinnvoll.

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 Codesequenzen 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`:

```
static String extractEmail(String line) {
    line = line.trim(); // find the first blank
    int i = line.indexOf(' ');
    if (i > 0)
        return line.substring(i).trim();
    else
        return null;
}
```

Dieses Codestück wurde mit

```
\begin{JavaCode}[numbers=none]
static String extractEmail(String line) {
    line = line.trim(); // find the first blank
    ...
}
\end{JavaCode}
```

erstellt (siehe Abschn. 4.6.1). In-line Programmstücke sollten maximal einige Zeilen lang sein und nach Möglichkeit nicht durch Seitenumbrüche geteilt werden.

Programmtexte als Float-Elemente

Sind längere Codesequenzen 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 `hgb.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}
```

Programm 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     int agent_velocity;
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

```

Wenn man möchte, kann man in diesem Fall die Caption auch unten anbringen (jedenfalls 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 4.1 und 4.2.

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 D

(Seite 82).⁹

⁹Grundsätzlich 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.

Kapitel 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 [Kopka98] und [mathmode09].

5.1 Mathematische Elemente im Fließtext

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

Der Nah-Unendlichkeitpunkt 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 `$$ \dots $$` erzeugt werden. Das Ergebnis wird zentriert, erhält jedoch keine Numerierung. 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

```
\begin{equation}
  f(k) = \frac{1}{N} \sum_{i=0}^{k-1} i^2 .
  \label{eq:MyFirstEquation}
\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 (5.1) dieser Gleichung (siehe dazu auch Abschn. 5.2.5). 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 **Interpunktion** (wie in Gl. 5.1 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 [Mermin89] und [Higham98].

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 [amslatex] zu greifen.

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

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:

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

$$f(i) = \begin{cases} 0 & \text{für } i = 0, \\ f(i-1) + f(i) & \text{für } i > 0. \end{cases} \quad (5.4)$$

mit folgenden Anweisungen erzeugt:

```
\begin{equation}
  f(i) =
  \begin{cases}
    0 & \& \text{für } \$i = 0\$}, \\
    f(i-1) + f(i) & \& \text{für } \$i > 0\$}.
  \end{cases}
\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 Interpunktion 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:

```

\begin{equation}
  \begin{pmatrix}
    x' \\
    y'
  \end{pmatrix}
=
  \begin{pmatrix}
    \cos \phi & -\sin \phi \\
    \sin \phi & \phantom{-}\cos \phi
  \end{pmatrix}
\cdot
  \begin{pmatrix}
    x \\
    y
  \end{pmatrix} ,
\end{equation}

```

Ein nützliches Detail darin ist das TeX-Makro `` (in Zeile 9), das sein Argument unsichtbar einfügt und hier als Platzhalter für das darüberliegende Minuszeichen verwendet wird. Alternativ zu `pmatrix` kann man mit der `bmatrix`-Umgebung Matrizen und Vektoren mit eckigen Klammern erzeugen. Zahlreiche weitere mathematische Konstrukte des `amsmath`-Pakets sind in [amslatex] 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 (5.2) abgeleitet werden kann ...“

Um Missverständnisse zu vermeiden, sollte man aber – vor allem in Texten mit nur wenigen mathematischen Elementen – „Gleichung 5.2“, „Gl. 5.2“ oder „Gl. (5.2)“ 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 `hagenberg`-Paket sind diese Symbole als Makros `\R`, `\Z`, `\N`, `\Cpx`, `\Q` (\mathbb{R} , \mathbb{Z} , \mathbb{N} , \mathbb{C} , \mathbb{Q}) mithilfe der *AMS Blackboard Fonts* definiert, z. B.:

$$x \in \mathbb{R}, k \in \mathbb{N}_0, z = (a + i \cdot b) \in \mathbb{C}.$$

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 das Textzeichen „x“!

5.3.3 Variable (Symbole) mit mehreren Zeichen

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

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

fälschlicherweise erzeugt durch

$$\text{\$Scalefactor \leftarrow Scalefactor^2 \cdot 1.5\$}$$

Dabei interpretiert LaTeX allerdings die Zeichenkette „Scalefactor“ als 11 einzelne, aufeinanderfolgende Symbole S, c, a, l, e, \dots und setzt dazwischen entsprechende Abstände. **Richtig** ist, diese Buchstaben mit `\mathit{...}` zu *einem* Symbol zusammenzufassen. Der Unterschied ist in diesem Fall deutlich sichtbar:

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

Grundsätzlich sollte man aber derart lange Symbolnamen aber ohnehin vermeiden und stattdessen möglichst kurze (gängige) 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

$$\sin \theta = \sin(\theta + 2\pi) \quad \leftarrow \quad \text{\$ \sin \theta = \sin(\theta + 2 \pi) \$}$$

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

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

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

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 `\$` und das Euro-Symbol (€) mit dem Makro `\euro`.³

5.3.6 Kommas in Dezimalzahlen (Mathematik-Modus)

LaTeX setzt im Mathematik-Modus (also innerhalb von `$$` oder in Gleichungen) nach dem angloamerikanischen 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. Das Komma wird in diesem Fall als **Satzzeichen** interpretiert und sieht dann so aus:

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

(man 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:

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

5.3.7 Mathematische 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.

³Das € Zeichen ist nicht im ursprünglichen LaTeX-Zeichensatz enthalten sondern wird mit dem `eurosym`-Paket erzeugt.

Algorithmus 5.1: Bikubische Interpolation in 2D. $w_{\text{cub}}()$ in Zeile 8 bezeichnet die eindimensionale kubische Interpolationsfunktion.

```

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

```

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. 5.1 wurde mit der Float-Umgebung `algorithm` und dem `algorithmicx`-Paket ausgeführt (Quellcode in Prog. 5.1).

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 [BurgerBurge06].

```

\begin{algorithm}
\caption{Bikubische Interpolation in 2D.
   $w_{\mathrm{cub}}()$  in Zeile \ref{alg:wcub} bezeichnet die
  eindimensionale kubische Interpolationsfunktion.}
\label{alg:Example}

\begin{algorithmic}[1] % [1] heißt alle Zeilen werden numeriert
\Procedure{BicubicInterpolation}{$I, x, y$} \Comment{$x, y \in \mathbb{R}$}
  \State Returns the interpolated value of the image  $I$ 
    at the continuous position  $(x, y)$ .

  \State  $\mathit{val} \leftarrow 0$ 

  \For{$j \leftarrow 0, \ldots, 3$} \Comment{iterate over 4 lines}
    \State  $v \leftarrow \lfloor y \rfloor - 1 + j$ 
    \State  $p \leftarrow 0$ 

    \For{$i \leftarrow 0, \ldots, 3$} \Comment{iterate over 4 columns}
      \State  $u \leftarrow \lfloor x \rfloor - 1 + i$ 
      \State  $p \leftarrow p + I(u, v) \cdot w_{\mathrm{cub}}(x - u)$ 
      \label{alg:wcub}
    \EndFor

    \State  $\mathit{val} \leftarrow \mathit{val} + p \cdot w_{\mathrm{cub}}(y - v)$ 
  \EndFor

  \State \Return  $\mathit{val}$ 
\EndProcedure
\end{algorithmic}
\end{algorithm}

```

Programm 5.1: Quellcode zu Algorithmus 5.1 (mit `algorithmicx`). Wie man sieht, kann man hier auch beliebig Leerzeilen verwenden, was die Lesbarkeit deutlich verbessert.

Kapitel 6

Umgang mit Literatur und anderen Quellen

Anmerkung: Der Titel dieses Kapitels ist absichtlich so lang geraten, dass er nicht mehr in die Kopfzeile der Seiten passt. In diesem Fall kann man in der `\chapter`-Anweisung als optionales Argument [...] einen verkürzten Text für die Kopfzeile (und das Inhaltsverzeichnis) angeben:

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

6.1 Allgemeines

Der richtige Umgang mit Quellen ist ein wesentliches Element bei der Erstellung wissenschaftlicher Arbeiten im Allgemeinen (s. auch Abschnitt 6.4). Für die Gestaltung von Quellenangaben sind unterschiedlichste Richtlinien in Gebrauch, bestimmt u. a. vom jeweiligen Fachgebiet oder Richtlinien von Verlagen und Hochschulen. Diese Vorlage sieht ein Schema vor, das in den naturwissenschaftlich-technischen Disziplinen üblich ist.¹ Technisch basiert dieser Teil auf BibTeX [Patashnik88] bzw. Biber² in Kombination mit dem Paket biblatex [Lehman2011].

Die Verwaltung von Quellen besteht grundsätzlich aus zwei Elementen: *Quellenverweise* im Text beziehen sich auf Einträge im *Quellenverzeichnis* (oder in mehreren Quellenverzeichnissen). Das Quellenverzeichnis ist eine Zusammenstellung aller verwendeten Quellen, typischerweise ganz am Ende des Dokuments. Wichtig ist, dass jeder Quellenverweis einen zugehörigen, eindeutigen Eintrag im Quellenverzeichnis aufweist und jedes Element im Quellenverzeichnis auch im Text referenziert wird.

¹Anpassungen an andere Formen sind relativ leicht möglich.

²Wird seit Version 2013/02/19 anstelle von bibtex verwendet (s. <http://biblatex-biber.sourceforge.net/>),

6.2 Quellenverweise

6.2.1 Das `\cite` Makro

Für Quellenverweise im laufenden Text verwendet man die Anweisung

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

Verweise ist eine durch Kommas getrennte Auflistung von Quellen-*Schlüsseln* zur Identifikation der entsprechenden Einträge im Quellenverzeichnis. Mit *Zusatztext* können Ergänzungstexte zum aktuellen Quellenverweis 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 **[Kopka98]**.

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

→ Mehr über *Styles* in **[Kopka98]**.

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

→ Die Angaben in **[Ears99]** sind aus meiner Sicht falsch.

Überholt sind auch `\cite{Ears99,Patashnik88,Duden97}`.

→ Überholt sind auch **[Ears99, Patashnik88, Duden97]**.

Die Sortierung der Angaben im letzten Beispiel erfolgt automatisch.

6.2.2 Häufige Fehler

Verweise außerhalb des Satzes

Quellenverweise sollten innerhalb oder am Ende eines Satzes (vor dem Punkt) stehen, nicht *außerhalb*:

Falsch: ... hier ist der Satz aus. **[Oetiker01]** Und jetzt geht es weiter ...

Richtig: ... hier ist der Satz aus **[Oetiker01]**. Und jetzt geht es weiter ...

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. Als Beispiel die folgende Passage aus **[Oetiker01]**:

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.³

³Man beachte die Verwendung von englischen Hochkommas innerhalb dieses Zitats.

Für das Zitat selbst sollte man übrigens das dafür vorgesehene Umgebung

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

verwenden, die 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!

6.3 Quellenverzeichnis

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

1. Das Quellenverzeichnis manuell zu formatieren (s. [Kopka98]).
2. Die Verwendung von BibTeX/Biber und einer zugehörigen Literaturdatenbank (s. [Kopka98]).

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.

6.3.1 Literaturdaten in BibTeX

BibTeX ist ein eigenständiges Programm, das aus einer „Literaturdatenbank“ (eine oder mehrere Textdateien von vorgegebener Struktur) ein für LaTeX geeignetes Quellenverzeichnis erzeugt. Literatur zur Verwendung von BibTeX findet man online, z. B. [Taylor96, Patashnik88]. Die BibTeX-Datei zu dieser Vorlage ist `literatur.bib` (im Hauptverzeichnis).

Man kann BibTeX-Dateien natürlich mit einem Texteditor manuell erstellen und für viele Literaturquellen sind bereits fertige BibTeX-Einträge online verfügbar. Dabei sollte man allerdings vorsichtig sein, denn diese Einträge sind (auch bei großen Institutionen und Verlagen) **häufig falsch oder syntaktisch fehlerhaft!** Man sollte sie daher nicht ungeprüft übernehmen und insbesondere die Endergebnisse genau kontrollieren. Darüber hinaus gibt es eigenständige Anwendungen zur Wartung von BibTeX-Verzeichnissen, wie beispielsweise *JabRef*.⁴

Dieses Dokument verwendet `biblatex` (Version 1.4 oder höher) in Verbindung mit dem Programm `biber`, das viele Unzulänglichkeiten des traditionellen BibTeX-Workflows behebt und dessen Möglichkeiten deutlich erweitert.⁵ Allerdings sind die in `biblatex` verwendeten Literaturdaten nicht

⁴<http://jabref.sourceforge.net>

⁵Tatsächlich ist `biblatex` die erste radikale (und längst notwendige) Überarbeitung des mittlerweile stark in die Jahre gekommenen BibTeX-Workflows. Zwar wird dabei BibTeX weiterhin für die Sortierung der Quellen verwendet, die Formatierung der Einträge und viele andere Elemente werden jedoch ausschließlich über LaTeX-Makros gesteuert.

mehr vollständig rückwärts-kompatibel zu BibTeX. Es ist daher in der Regel notwendig, bestehende oder aus Online-Quellen übernommene BibTeX-Daten manuell zu überarbeiten (s. auch Abschnitt 6.3.7).

In dieser Vorlage sind die Schnittstellen zu `biblatex` weitgehend in der Datei `hgbthesis.cls` verpackt. Die typische Verwendung in der LaTeX-Hauptdatei sieht folgendermaßen aus:

```
\documentclass[diplom,german]{hgbthesis}
...
\bibliography{literatur}
...
\begin{document}
...
\MakeBibliography{Quellenverzeichnis}
\end{document}
```

In der „Präambel“ (Zeile 3) wird mit `\bibliography{literatur}` auf eine (modifizierte) BibTeX-Datei `literatur.bib` verwiesen.⁶ Falls man mehrere BibTeX-Dateien verwendet, kann man sie in der gleichen Form angeben.

Die Anweisung `\MakeBibliography{..}` am Ende des Dokuments (Zeile 7) besorgt die Ausgabe des Quellenverzeichnisses, hier mit dem Titel „Quellenverzeichnis“. Dabei sind zwei Varianten möglich:

`\MakeBibliography{title}`

Erzeugt ein in mehrere *Kategorien* (s. Abschnitt 6.3.2) geteiltes Quellenverzeichnis mit der Hauptüberschrift `title`. Diese Variante wird im vorliegenden Dokument verwendet.

`\MakeBibliography[nosplit]{title}`

Erzeugt ein traditionelles *einteiliges* Quellenverzeichnis mit der Überschrift `title`.

Die Überschrift `title` scheint in beiden Fällen ohne Nummerierung im Inhaltsverzeichnis auf (auf `chapter`-Ebene). Bei geteiltem Quellenverzeichnis werden die Überschriften der einzelnen Abschnitte auf `section`-Ebene eingetragen.

6.3.2 Kategorien von Quellenangaben

Für geteilte Quellenverzeichnisse sind in dieser Vorlage folgende drei Kategorien vorgesehen (s. Tabelle 6.1):⁷

literature – für alle Quellen, die in gedruckter Form vorliegen;

⁶Das Makro `\bibliography` ist eigentlich ein Relikt aus BibTeX und wird in `biblatex` durch die Anweisung `\addbibresource` ersetzt. Beide Anweisungen sind gleichwertig, allerdings wird nur mit `\bibliography` die zugehörige `.bib`-Datei im Fileverzeichnis von TeXnicCenter angezeigt.

⁷Diese sind in der Datei `hgbthesis.cls` definiert. Allfällige Änderungen sowie die Definition zusätzlicher Kategorien sind bei Bedarf relativ leicht möglich.

Tabelle 6.1: Quellenarten und empfohlene BibTeX-Eintragstypen.

Kategorie <i>literature</i>	@ <i>type</i>	Seite
Buch (Textbuch, Monographie)	@book	54
Sammelband (Hrsg. + mehrere Autoren)	@incollection	55
Konferenz-, Tagungsband	@inproceedings	56
Beitrag in Zeitschrift, Journal	@article	56
Bachelor-, Master-, Diplomarbeit, Dissertation	@thesis	57
Technischer Bericht, Laborbericht	@techreport	58
Handbuch, Produktbeschreibung, Norm	@manual	59
Gesetzestext, Verordnung etc.	@misc	60
Kategorie <i>avmedia</i>		
Audio (CD)	@audio	60
Video (auf VHS, DVD, Blu-ray Disk)	@video	61
Film (Kino)	@movie	61
Computer Game, Softwareprodukt	@software	62
Kategorie <i>online</i>		
Webseite, Wiki-Eintrag, Blog etc.	@online	63
Online-Video, Bild, Grafik	@online	63

avmedia – für Filme, audio-visuelle Medien und Computer Games (auf DVD, CD, usw.);

online – für Werke, die *ausschließlich* online verfügbar sind.

Jedes Quellenobjekt wird aufgrund des angegebenen BibTeX-Eintragstyps **@type** automatisch einer dieser Kategorien zugeordnet (s. Tabelle 6.2). Angeführt sind hier nur die wichtigsten Eintragstypen, die allerdings die meisten Fälle in der Praxis abdecken sollten und nachfolgend durch Beispiele erläutert sind. Alle nicht explizit angegebenen Einträge werden grundsätzlich in der Kategorie *literature* ausgegeben.

6.3.3 Gedruckte Quellen (*literature*)

Diese Kategorie umfasst alle Werke, die in gedruckter Form publiziert wurden, also beispielsweise in Büchern, Konferenzbänden, Zeitschriftenartikeln, Diplomarbeiten usw. In den folgenden Beispielen ist jeweils der BibTeX-Eintrag in der Datei `literatur.bib` angegeben, gefolgt vom zugehörigen Ergebnis im Quellenverzeichnis.

@book

Ein einbändiges Buch (Monographie), das von einem Autor oder mehreren Autoren zur Gänze gemeinsam verfasst und (typischerweise) von einem Verlag herausgegeben wurde.

Tabelle 6.2: Kategorien von Quellenangaben und zugehörige BibTeX-Eintragstypen. Bei geteiltem Quellenverzeichnis werden die Einträge jeder Kategorie in einem eigenen Abschnitt gesammelt. Grau gekennzeichnete Elemente sind Synonyme für die jeweils darüber stehenden Typen.

<i>literature</i>	<i>avmedia</i>	<i>online</i>
@book	@audio	@online
@incollection	@music	@electronic
@inproceedings	@video	@www
@article	@movie	
@thesis	@software	
@techreport		
@manual		
@misc		
...		

```
@book{BurgerBurge06,
  author={Burger, Wilhelm and Burge, Mark},
  title={Digitale Bildverarbeitung --
    Eine Einführung mit Java und ImageJ},
  publisher={Springer-Verlag},
  address={Heidelberg},
  edition={2},
  year={2005},
  hyphenation={german}
}
```

[BurgerBurge06] Wilhelm Burger und Mark Burge. *Digitale Bildverarbeitung – Eine Einführung mit Java und ImageJ*. 2. Aufl. Heidelberg: Springer-Verlag, 2006.

Hinweis: Die Auflagennummer (**edition**) wird üblicherweise nur angegeben, wenn es *mehr* als eine Ausgabe gibt – also insbesondere **nicht für die 1. Auflage**, wenn diese die einzige ist! **ISBN-Nummern** sollte man auch getrost **weglassen**.

@incollection

Ein in sich abgeschlossener und mit einem eigenen Titel versehener Beitrag eines oder mehrerer Autoren in einem Buch oder Sammelband. Dabei ist **title** der Titel des Beitrags, **booktitle** der Titel des Sammelbands und **editor** der Name des Herausgebers.

```
@incollection{Ears99,
  author={Burge, Mark and Burger, Wilhelm},
  title={Ear Biometrics},
  booktitle={Biometrics: Personal Identification in Networked
    Society},
```

```

publisher={Kluwer Academic Publishers},
year={1999},
address={Boston},
editor={Jain, Anil K. and Bolle, Ruud and Pankanti, Sharath},
chapter={13},
pages={273-285},
hyphenation={english}
}

```

- [Ears99] Mark Burge und Wilhelm Burger. „Ear Biometrics“. In: *Biometrics: Personal Identification in Networked Society*. Hrsg. von Anil K. Jain, Bolle Ruud und Pankanti Sharath. Boston: Kluwer Academic Publishers, 1999. Kap. 13, S. 273–285.

@inproceedings

Konferenzbeitrag, individueller Beitrag in einem Tagungsband. Man beachte die Verwendung des neuen Felds `venue` zur Angabe des Tagungsorts und `address` für den Ort der Publikation (des Verlags).

```

@inproceedings{Burger87,
  author={Burger, Wilhelm and Bhanu, Bir},
  title={Qualitative Motion Understanding},
  booktitle={Proceedings of the Intl.\ Joint Conference on
    Artificial Intelligence},
  year={1987},
  month={5},
  editor={McDermott, John P.},
  venue={Mailand},
  publisher={Morgan Kaufmann Publishers},
  address={San Francisco},
  pages={819-821},
  hyphenation={english}
}

```

- [Burger87] Wilhelm Burger und Bir Bhanu. „Qualitative Motion Understanding“. In: *Proceedings of the Intl. Joint Conference on Artificial Intelligence*. (Mailand). Hrsg. von John P. McDermott. San Francisco: Morgan Kaufmann Publishers, Mai 1987, S. 819–821.

@article

Beitrag in einer Zeitschrift, einem wissenschaftlichen Journal oder einer Tageszeitung. Dabei steht `volume` üblicherweise für den Jahrgang und `number` für die Nummer innerhalb des Jahrgangs. Den Zeitschriftennamen (`journal` oder `journaltitle`) sollte man nur in begründeten Fällen abkürzen, um Missverständnisse zu vermeiden.

```

@article{Mermin89,
  author={Mermin, Nathaniel David},
  title={What's wrong with these equations?},

```

```

journal={Physics Today},
volume={42},
number={10},
year={1989},
pages={9-11},
hyphenation={english}
}

```

[**Mermin89**] Nathaniel David Mermin. „What’s wrong with these equations?“ In: *Physics Today* 42.10 (1989), S. 9–11.

Hinweis: Die Angabe einer Ausgabe für *mehrere* Monate ist in **biblatex** nicht mehr über das Feld **month** möglich, denn dieses darf nur mehr *einen* Wert enthalten. In diesem Fall kann man das **issue**-Feld verwenden (z. B. **issue={Mai/Juni}**) im BibTeX-Eintrag zu [**Guttman01**] und das **number**-Feld weglassen.

@thesis

Dieser (neue) Eintragstyp kann allgemein für akademische Abschlussarbeiten verwendet werden. Er ersetzt insbesondere die bekannten BibTeX-Einträge **@phdthesis** (für Dissertationen) sowie **@mastersthesis** (für Diplom- und Masterarbeiten), die allerdings weiterhin verwendet werden können. Zusätzlich ist damit etwa auch die Angabe von Bachelorarbeiten möglich.

Dissertation (Doktorarbeit):

```

@thesis{Eberl87,
author={Eberl, Gerhard},
title={Automatischer Landeanflug durch Rechnersehen},
type={phdthesis},
year={1987},
month={8},
institution={Universität der Bundeswehr, Fakultät für Raum- und
Luftfahrttechnik},
address={München},
hyphenation={german}
}

```

[**Eberl87**] Gerhard Eberl. „Automatischer Landeanflug durch Rechnersehen“. Diss. München: Universität der Bundeswehr, Fakultät für Raum- und Luftfahrttechnik, Aug. 1987.

Magister- oder Masterarbeit:

Analog zur Dissertation (s. oben), allerdings mit **type={mathesis}**.

Diplomarbeit:

Analog zur Dissertation (s. oben), allerdings mit **type={Diplomarbeit}**:

```
@thesis{Artner07,
  author={Artner, Nicole Maria},
  title={Analyse und Reimplementierung des Mean-Shift Tracking-
    Verfahrens},
  type={Diplomarbeit},
  year={2007},
  month={7},
  institution={University of Applied Sciences Upper Austria,
    Digitale Medien},
  address={Hagenberg, Austria},
  url={http://theses.fh-hagenberg.at/thesis/Artner07},
  hyphenation={german}
}
```

[**Artner07**] Nicole Maria Artner. „Analyse und Reimplementierung des Mean-Shift Tracking-Verfahrens“. Diplomarbeit. Hagenberg, Austria: Upper Austria University of Applied Sciences, Digitale Medien, Juli 2007. URL: <http://theses.fh-hagenberg.at/thesis/Artner07>.

Der Inhalt des Felds `url={..}` wird dabei automatisch und ohne zusätzliche Kennzeichnung als URL gesetzt (mit dem `\url{..}` Makro).

Bachelorarbeit:

Bachelorarbeiten gelten in der Regel zwar nicht als „richtige“ Publikationen, bei Bedarf muss man sie aber dennoch referenzieren können.

```
@thesis{Bacher04,
  author={Bacher, Florian},
  title={Interaktionsmöglichkeiten mit Bildschirmen und großflä-
    chigen Projektionen},
  type={Bachelorarbeit},
  year={2004},
  month={6},
  institution={Upper Austria University of Applied Sciences,
    Medientechnik und {-design}},
  address={Hagenberg, Austria},
  hyphenation={german},
  keywords={bachelorarbeitexample1}
}
```

[**Bacher04**] Florian Bacher. „Interaktionsmöglichkeiten mit Bildschirmen und großflächigen Projektionen“. Bachelorarbeit. Hagenberg, Austria: Upper Austria University of Applied Sciences, Medientechnik und -design, Juni 2004.

@techreport

Das sind typischerweise nummerierte Berichte (*technical reports*) aus Unternehmen, Hochschulinstituten oder Forschungsprojekten. Wichtig ist, dass die herausgebende Organisationseinheit (Firma, Institut, Fakultät etc.) und

Adresse angegeben wird. Sinnvollerweise wird man auch den zugehörigen URL angeben, sofern vorhanden.

```
@techreport{Drake48,
  author={Drake, Huber M. and McLaughlin, Milton D. and Goodman,
    Harold R.},
  title={Results obtained during accelerated transonic tests of
    the {Bell} {XS-1} airplane in flights to a {MACH} number of
    0.92},
  institution={NASA Dryden Flight Research Center},
  year={1948},
  month={1},
  address={Edwards, CA},
  number={NACA-RM-L8A05A},
  url={http://www.nasa.gov/centers/dryden/pdf/...05A.pdf},
  hyphenation={english}
}
```

- [**Drake48**] Huber M. Drake, Milton D. McLaughlin und Harold R. Goodman. *Results obtained during accelerated transonic tests of the Bell XS-1 airplane in flights to a MACH number of 0.92*. Techn. Ber. NACA-RM-L8A05A. Edwards, CA: NASA Dryden Flight Research Center, Jan. 1948. URL: http://www.nasa.gov/centers/dryden/pdf/87528main_RM-L8A05A.pdf.

@manual

Dieser Publikationstyp bietet sich für Dokumente an, bei denen üblicherweise kein Autor genannt ist, wie etwa bei Produktbeschreibungen von Herstellern, Anleitungen, Präsentationen, Normen, White Papers usw.

```
@manual{amsl doc02,
  author={{American Mathematical Society}},
  title={User's Guide for the {\tt amsmath} Package},
  year={2002},
  version={2.0},
  url={ftp://ftp.ams.org/pub/tex/doc/amsmath/amsl doc.pdf},
  hyphenation={english}
}
```

- [**amsl doc02**] American Mathematical Society. *User's Guide for the amsmath Package*. Version 2.0. 2002. URL: <ftp://ftp.ams.org/pub/tex/doc/amsmath/amsl doc.pdf>.

In obigem Beispiel ist folgendes zu beachten: Da kein Autor bekannt ist, wird hier der Namen des *Unternehmens* oder der *Institution* im **author**-Feld angegeben, allerdings innerhalb einer **zusätzlichen Klammer** {. .}, damit das Argument nicht fälschlicherweise als *Vorname+Nachname* interpretiert wird.⁸

⁸Im Unterschied zu BibTeX wird in biblatex bei @manual-Einträgen das Feld **organization** nicht als Ersatz für **author** akzeptiert.

@misc

Sollte man mit den bisher angeführten Eintragungstypen für gedruckte Publikationen nicht das Auslangen finden, sollte man sich zunächst die weiteren (hier nicht näher beschriebenen) Typen im **biblatex**-Handbuch [Lehman2011] ansehen, beispielsweise **@collection** für einen Sammelband als Ganzes (also nicht nur ein Beitrag darin) oder **@patent** für ein Patent oder eine Patentanmeldung.

Wenn nichts davon passt, dann kann man auf den Typ **@misc** zurückgreifen, der ein Textfeld **howpublished** vorsieht, in dem man die Art der Publikation individuell angeben kann. Das folgende Beispiel zeigt die Anwendung für einen Gesetzestext (s. auch [FhStG93] und [EuRichtlinie2000]).

```
@misc{OoeRaumordnungsgesetz94,
  title={Oberösterreichisches Raumordnungsgesetz 1994},
  howpublished={LGBI 1994/114 idF 1995/93},
  url={http://www.ris.bka.gv.at/Dokumente/Lr00/...538.pdf},
  hyphenation={german}
}
```

OoeRaumordnungsgesetz94] *Oberösterreichisches Raumordnungsgesetz 1994*. LGBI 1994/114 idF 1995/93. URL: <http://www.ris.bka.gv.at/Dokumente/Lr00/LOO40007538/LOO40007538.pdf>.

6.3.4 Filme und audio-visuelle Medien (*avmedia*)

Diese Kategorie ist dazu vorgesehen, audio-visuelle Produktionen wie Filme, Tonaufzeichnungen, Audio-CDs, DVDs, VHS-Kassetten usw. zu erfassen. Damit gemeint sind Werke, die in physischer (jedoch nicht in gedruckter) Form veröffentlicht wurden. Nicht gemeint sind damit audio-visuelle Werke (Tonaufnahmen, Bilder, Videos) die ausschließlich online verfügbar sind – diese sollten mit einem Elementtyp **@online** (s. Tabelle 6.2 und Abschnitt 6.3.5) ausgezeichnet werden.

Die nachfolgend beschriebenen Typen **@audio**, **@video** und **@movie** sind *keine* BibTeX-Standardtypen. Sie sind aber in **biblatex** vorgesehen (und implizit durch **@misc** ersetzt) und werden hier empfohlen, um die automatische Gliederung des Quellenverzeichnisses zu ermöglichen.

@audio

Hier ein Beispiel für die Spezifikation einer Audio-CD:

```
@audio{Zappa95,
  author={Zappa, Frank},
  title={Freak Out},
  howpublished={Audio-CD},
  year={1995},
  month={5},
  note={Rykodisc, New York},
}
```

```
hyphenation={english},
keywords={audioexample1}
}
```

[Zappa95] Frank Zappa. *Freak Out*. Audio-CD. Rykodisc, New York. Mai 1995.

Anstelle von `howpublished={Audio-CD}` könnte man auch `type={audiocd}` verwenden.

@video

Hier ein Beispiel für die Spezifikation einer DVD-Edition:

```
@video{Futurama99,
author={Groening, Matt},
title={Futurama -- Season 1 Collection},
howpublished={DVD},
year={2002},
month={2},
note={Twentieth Century Fox Home Entertainment},
hyphenation={english}
}
```

[Futurama99] Matt Groening. *Futurama – Season 1 Collection*. DVD. Twentieth Century Fox Home Entertainment. Feb. 2002.

In diesem Fall ist das angegebene Datum der *Erscheinungstermin*. Falls kein eindeutiger Autor namhaft gemacht werden kann, lässt man das `author`-Feld weg und verpackt die entsprechenden Angaben im `note`-Feld, wie im nachfolgenden Beispiel gezeigt.

@movie

Dieser Eintragstyp ist für Filme reserviert. Hier wird von vornherein *kein* Autor angegeben, weil dieser bei einer Filmproduktion i. Allg. nicht eindeutig zu benennen ist. Im folgenden Beispiel (s. auch [Psycho]) sind die betreffenden Daten im `note`-Feld angegeben:⁹

```
@movie{Nosferatu,
title={Nosferatu -- A Symphony of Horrors},
howpublished={Film},
year={1922},
note={Drehbuch/Regie: F. W. Murnau. Mit Max Schreck, Gustav
von Wangenheim, Greta Schröder.},
hyphenation={english}
}
```

[Nosferatu] *Nosferatu – A Symphony of Horrors*. Film. Drehbuch/Regie: F. W. Murnau. Mit Max Schreck, Gustav von Wangenheim, Greta Schröder. 1922.

⁹Übrigens achtet `biblatex` netterweise darauf, dass der Punkt am Ende des `note`-Texts in der Ausgabe nicht verdoppelt wird.

Die Angabe `howpublished={Film}` ist hier sinnvoll, um die Verwechslung mit einem (möglicherweise gleichnamigen) Buch auszuschließen.

@software

Dieser Eintragstyp ist insbesondere für Computerspiele geeignet (in Ermangelung eines eigenen Eintragstyps). Hier ein konkretes Beispiel [**LegendOfZelda98**]:

```
@software{LegendOfZelda98,
  author={Miyamoto, Shigeru and Aonuma, Eiji and Koizumi,
    Yoshiaki},
  title={The Legend of Zelda: Ocarina of Time},
  howpublished={N64-Spielmodul},
  publisher={Nintendo},
  year={1998},
  hyphenation={english}
}
```

Zeitangaben zu Musikaufnahmen und Filmen

Einen Verweis auf eine bestimmten Stelle in einem Musikstück oder Film kann man ähnlich ausführen wie die Seitenangabe in einem Druckwerk. Besonders legendär (und häufig parodiert) ist beispielsweise die Duschszene in *Psycho* [**Psycho**]. Alternativ zur einfachen Zeitangabe „T=hh:mm:ss“ könnte man eine bestimmte Stelle auch auf den Frame genau durch den zugehörigen *Timecode* „TC=hh:mm:ss:ff“ angeben, z. B. [**Psycho**] für Frame ff=12.

6.3.5 Online-Quellen (*online*)

Bei Verweisen auf Online-Ressourcen sind grundsätzlich drei Fälle zu unterscheiden:

- A. Man möchte allgemein auf eine Webseite verweisen, etwa auf die „Professional Broadcast“ Seite von Panasonic.¹⁰ In diesem Fall wird nicht auf ein konkretes „Werk“ verwiesen und daher erfolgt *keine* Aufnahme ins Quellenverzeichnis. Stattdessen genügt eine einfache Fußnote mit `\footnote{\url{...}}`, wie im vorigen Satz gezeigt.
- B. Ein gedrucktes oder audio-visuelles Werk (s. Abschnitte 6.3.3–6.3.4) ist *zusätzlich* auch online verfügbar. In diesem Fall ist die Primärpublikation aber *nicht* „online“ und es genügt, ggfs. den zugehörigen Link im `url`-Feld anzugeben, das bei jedem Eintragstyp zulässig ist.
- C. Es handelt sich im weitesten Sinn um ein Werk, das aber *ausschließlich* online verfügbar ist, wie z. B. ein Wiki-Eintrag, ein digitales Bild, eine Software, ein YouTube-Video oder ein Blog-Eintrag. Die Kategorie *online* ist genau (und *nur*) für diese Art von Quellen vorgesehen.

¹⁰<http://www.panasonic-broadcast.de>

Beispiel: Wiki-Eintrag

Durch den Umfang und die steigende Qualität dieser Einträge erscheint die Aufnahme in das Quellenverzeichnis durchaus berechtigt. Beispielsweise bezeichnet man als „Reliquienschrein“ einen Schrein, in dem die Reliquien eines oder mehrerer Heiliger aufbewahrt werden [**WikiReliquienschrein**].

```
@online{WikiReliquienschrein,
  url={http://de.wikipedia.org/wiki/Reliquienschrein},
  urldate={2011-06-19}
}
```

[**WikiReliquienschrein**] URL: <http://de.wikipedia.org/wiki/Reliquienschrein> (besucht am 19.06.2011).

In diesem Fall besteht die Quellenangabe praktisch nur mehr aus dem URL. Durch die (optionale) Angabe von `urldate` (im YYYY-MM-DD Format) wird automatisch die Information eingefügt, wann das Online-Dokument eingesehen wurde.

Beispiel: online Video

```
@online{HistoryOfComputers2008,
  title={History of Computers},
  year={2008},
  url={http://www.youtube.com/watch?v=LvKxJ3bQRKE},
  hyphenation={english}
}
```

[**HistoryOfComputers2008**] *History of Computers*. 2008. URL: <http://www.youtube.com/watch?v=LvKxJ3bQRKE>.

Bis auf das Feld `url` können alle Angaben weggelassen werden. Die Angabe von weiteren Details (z. B. `author`) ist natürlich ebenso möglich.

Beispiel: Bildquelle

In Abbildungen wird sehr häufig fremdes Bildmaterial verwendet, dessen Herkunft natürlich in jedem Fall angegeben werden sollte. Die Angabe von URLs unmittelbar in den Captions von Abbildungen ist problematisch, weil sie meistens für das Schriftbild ziemlich störend sind. Einfacher ist es, auch einzelne Bilder und Grafiken ins Quellenverzeichnis aufzunehmen und wie üblich mit dem `\cite{..}` Kommando darauf zu verweisen. Beispiele dazu sind die Angaben zu Abb. 4.1–4.2.

```
@online{IBM360,
  url={http://www.plyojump.com/classes/mainframe_era.html}
}
```

[**IBM360**] URL: http://www.plyojump.com/classes/mainframe_era.html.

6.3.6 Elektronische Datenträger als Ergänzung zur Arbeit

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 die zugehörige Quelle im Quellenverzeichnis mit einem entsprechenden Verweis im **note**-Feld – z. B. „Kopie auf CD-ROM (Datei xyz.pdf)“ – zu versehen. Für die Angabe von solchen Dateinamen, die nicht als Online-Link zu öffnen sind, ist übrigens die Verwendung von `\nolinkurl{...}` anstelle von `\url{...}` zu empfehlen.

6.3.7 Tipps zur Erstellung von BibTeX-Dateien

month-Attribut

Das **month**-Attribut ist in **biblatex** (im Unterschied zu BibTeX) numerisch und wird beispielsweise einfach in der Form **month={8}** (für den Monat August) angegeben.

language/hyphenation-Attribut

Das **language** Attribut ermöglichte im Zusammenhang mit dem **babelbib**-Paket den korrekten Satz mehrsprachiger Quellenverzeichnisse. **babelbib** wird in dieser Vorlage *nicht* mehr verwendet, sondern durch **biblatex** [Lehman2011] ersetzt. Hier wird allerdings das **language**-Feld anders interpretiert und stattdessen das **hyphenation**-Feld verwendet, das man nach Möglichkeit bei jedem Quelleneintrag angeben sollte, also beispielsweise

`hyphenation={german}` oder `hyphenation={english}`

für ein deutsch- bzw. englischsprachiges Dokument.

edition-Attribut

Im Unterschied zu BibTeX ist mit **biblatex** auch das **edition**-Feld numerisch. Es ist lediglich die Nummer selbst anzugeben, also etwa **edition={3}** bei einer dritten Auflage. Die richtige Interpunktion in der Quellenangabe wird in Abhängigkeit von der Spracheinstellung automatisch hinzugefügt („3. Auflage“ bzw. „3rd edition“).

Wie bereits auf Seite 54 angemerkt, sollte im Fall einer **1. Auflage** das **edition**-Feld **nicht** verwendet werden, wenn es ohnehin keine andere Auflage gibt!

Übernahme von fertigen BibTeX-Einträgen

Viele Verlage und Literatur-Broker bieten fertige BibTeX-Einträge zum Download an. Dabei ist jedoch größte Vorsicht geboten, denn diese Einträge sind

häufig unvollständig oder syntaktisch fehlerhaft! Sie sollten bei der Übernahme *immer* auf Korrektheit überprüft werden! Besonderes sollte man dabei auf die korrekte Angabe der Namen, also `author = {Nachname1, Vorname1a Vorname1b and Nachname2, Vorname2a ...}`¹¹ achten, sowie die Angabe von `volume`, `number` und `pages`. Die Namen von Konferenzen sind oft völlig falsch (auch bei ACM und IEEE). ISBN- und ISSN-Nummern sind entbehrlich, man kann sie getrost weglassen. Gleiches gilt für DOI-Nummern – aber das möge der Betreuer entscheiden.

Listing aller Quellen

Durch die Anweisung `\nocite{*}` – an beliebiger Stelle im Dokument platziert – werden *alle* bestehenden Einträge der BibTeX-Datei im Quellenverzeichnis 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!

Ergänzungen zu biblatex

`biblatex` gibt bei der Verarbeitung dieses Dokuments einige Warnungen aus („Package biblatex Warning“), die man in der Regel ignorieren kann:

```
Package biblatex Warning: No driver for entry type 'video'.
(biblatex)                Using fallback driver ...
```

Für die neuen Eintragstypen `audio`, `video` und `movie` sind derzeit noch keine eigenen „driver“ hinterlegt. Stattdessen wird im Hintergrund der `misc`-Typ als „fallback“ verwendet.

Wenn eine der drei Kategorien des Quellenverzeichnisses (s. Abschnitt 6.3.2) ohne Eintrag ist, wird eine Warnung „Empty bibliography on input line ...“ ausgegeben, die man auch getrost ignorieren kann.

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 Hochschulwesen, hat sich aber durch die breite Verfügbarkeit elektronischer Quellen in den letzten Jahren dramatisch verstärkt und wird keineswegs als Kavaliersdelikt betrachtet. Viele Hochschulen bedienen sich als Gegenmaßnahme heute ebenfalls elektronischer Hilfsmittel (die den Studierenden zum Teil nicht zugänglich sind),

¹¹Das ist vor allem bei mehrteiligen Nachnamen wichtig, weil in diesem Fall Vor- und Nachnamen nicht korrekt zugeordnet werden können, z.B. `author = {van Beethoven, Ludwig and ter Linden, Jaap}` für ein hypothetisches Werk von Ludwig van Beethoven und Jaap ter Linden.

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 akademischen Grades 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.

Kapitel 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. Zur Arbeit mit dem Sumatra PDF-Viewer unter Windows sind entsprechende Ausgabeprofile für TexNicCenter vorbereitet, die aus der Datei `_tc_output_profile_sumatra_utf8.tco` importiert werden können.

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

Zunächst sollte man sicherstellen, dass die in der fertigen PDF-Datei eingestellte Papiergröße tatsächlich **A4** ist! Das geht z. B. mit *Adobe Acrobat* oder *SumatraPDF* über **File** → **Properties**, wo die Papiergröße des Dokuments angezeigt wird:

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

Richtig: A4 = $8,27 \times 11,69$ in bzw. $21,0 \times 29,7$ cm.

Falls das nicht stimmt, ist vermutlich irgendwo im Workflow versehentlich **Letter** als Papierformat eingestellt,

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 Diplomarbeit² ist in fest gebundener Form einzureichen.³ Dabei ist eine Bindung zu verwenden, die das Ausfallen von einzelnen Seiten nachhaltig verhindert, z. B. 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

²Für **Bachelorarbeiten** genügt, je nach Vorgaben des Studiengangs, meist eine einfache Bindung (Copsyshop oder Bibliothek).

³An der FH Hagenberg ist bei Diplomarbeiten zumindest eines der Exemplare *ungebunden* abzugeben – dieses wird später von einem Buchbinder in einheitlicher Form gebunden und verbleibt danach in der Bibliothek. Datenträger sind bei diesem Exemplar lose und *ohne* Aufkleber (jedoch beschriftet) beizulegen.

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.⁴

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 Label⁵ 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 B).

⁴Auch Bilder und Grafiken könnten in elektronischer Form nützlich sein, die LaTeX- oder Word-Dateien sind hingegen überflüssig.

⁵Nicht beim lose abgegebenen Bibliotheksexemplar – dieses erhält ein standardisiertes Label durch die Bibliothek.

Kapitel 8

Schlussbemerkungen¹

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?

8.1 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.

8.2 Checkliste

Abschließend noch eine kurze Liste der wichtigsten Punkte, an denen erfahrungsgemäß die häufigsten Fehler auftreten (Tab. 8.1).

¹Diese Anmerkung dient nur dazu, die (in seltenen Fällen sinnvolle) Verwendung von Fußnoten bei Überschriften zu demonstrieren.

Tabelle 8.1: Checkliste. Diese Punkte bilden auch die Grundlage der routinemäßigen Formbegutachtung in Hagenberg.

- ☐ **Titelseite:** Länge des Titels (Zeilenumbrüche), Name, Studiengang, Datum.
- ☐ **Erklärung:** vollständig Unterschrift.
- ☐ **Inhaltsverzeichnis:** balancierte 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.
- ☐ **Interpunktion:** 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.

Anhang A

Technische Informationen

A.1 Aktuelle Dateiversionen

Datum	Datei
2014/11/05	hgbthesis.cls
2014/11/05	hgb.sty

A.2 Details zur aktuellen Version

Das ist eine völlig überarbeitete Version der DA/BA-Vorlage, die UTF-8 kodierten Dateien vorsieht und ausschließlich im PDF-Modus arbeitet. Der „klassische“ DVI-PS-PDF-Modus wird somit nicht mehr unterstützt!

A.2.1 Allgemeine technische Voraussetzungen

Eine aktuelle LaTeX-Installation mit

- Texteditor für UTF-8 kodierte (Unicode) Dateien,
- **biber**-Programm (BibTeX-Ersatz, Version ≥ 1.5),
- **biblatex**-Paket (Version ≥ 2.5 , 2013/01/10),
- Latin Modern Schriften (Paket **lmodern**).¹

A.2.2 Verwendung unter Windows

Eine typische Installation unter Windows sieht folgendermaßen aus (s. auch Abschnitt 3.1.1):

1. **MikTeX 2.9**² (zurzeit am einfachsten die 32-Bit Version, da nur diese das Programm **biber.exe** bereits enthält),

¹<http://www.ctan.org/pkg/lm>, <http://www.tug.dk/FontCatalogue/lmodern>

²www.miktex.org – **Achtung:** Generell wird die **Komplettinstallation** von MikTeX („Complete MiKTeX“) empfohlen, da diese bereits alle notwendigen Zusatzpakete und Schriftdateien enthält! Bei der Installation ist darauf zu achten, dass die automatische Installation erforderlicher Packages durch „*Install missing packages on-the-fly*: = Yes“

2. **TeXnicCenter 2.0**³ (Editor-Umgebung, unterstützt UTF-8),
3. **SumatraPDF**⁴ (PDF-Viewer),

Ein passendes TeXnicCenter-Profil für MikTeX, Biber und Sumatra ist in diesem Paket enthalten (Datei `_tc_output_profile_sumatra_utf8.tco`). Dieses sollte man zuerst über **Build** → **Define Output Profiles** in TeXnicCenter importieren. **Achtung:** Alle neu angelegten `.tex`-Dateien sollten in UTF-8 Kodierung gespeichert werden!

A.2.3 Verwendung unter Mac OS

Diese Version sollte insbesondere mit *MacTeX* problemlos laufen (s. auch Abschnitt 3.1.1):

1. *MacTeX* (2012 oder höher).
2. Die Zeichenkodierung des Editors sollte auf UTF-8 eingestellt sein.
3. Als Engine (vergleichbar mit den Ausgabeprofilen in TeXnicCenter) sollte *LaTeXmk* verwendet werden. Dieses Perl-Skript erkennt automatisch, wie viele Aufrufe von *pdfLaTeX* und *Biber* nötig sind. Die Ausgabeprofile *LaTeX* oder *pdfLaTeX* hingegen müssen mehrmals aufgerufen werden, zudem werden hierbei auch die Literaturdaten nicht verarbeitet. Dazu müsste extra die *Biber*-Engine aufgerufen werden, die jedoch noch nicht in allen Editoren vorhanden ist.

ermöglicht wird (NICHT „*Ask me first*“)! Außerdem ist zu empfehlen, unmittelbar nach der Installation von MikTeX mit dem Programm MikTeX → **Maintenance** → **Update** und **Package Manager** ein Update der installierten Pakete durchzuführen.

³<http://www.texniccenter.org/>

⁴<http://blog.kowalczyk.info/software/sumatrapdf/>

Anhang B

Inhalt der CD-ROM/DVD

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

B.1 PDF-Dateien

Pfad: /

_DaBa.pdf	Diplom- oder Bachelorarbeit mit Instruktionen (Gesamtdokument)
_PrBericht.pdf	Praktikumsbericht (verkürzte Version der Bachelorarbeit)

B.2 LaTeX-Dateien

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

Pfad: /

_DaBa.tex	Diplom-/Bachelorarbeit (Hauptdokument)
_PrBericht.tex	Praktikumsbericht (verkürzte Version der Bachelorarbeit)
vorwort.tex	Vorwort
kurzfassung.tex	Kurzfassung
abstract.tex	Abstract
einleitung.tex	Kapitel 1

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

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 (Liste der Änderungen)
anhang_d.tex	Anhang D (LaTeX-Quellcode)
messbox.tex	Messbox zur Druckkontrolle
literatur.bib	Literatur-Datenbank (BibTeX-File)

B.3 Style/Class-Dateien

Pfad: /

hgbthesis.cls	LaTeX Class-Datei für Master- und Bachelorarbeiten
hgbtermreport.cls	LaTeX Class-Datei für Semesterberichte
hgb.sty	LaTeX Style-Datei für alle Hagenberg-Dokumente

B.4 Sonstiges

Pfad: /images

*.ai	Original Adobe Illustrator-Dateien
*.fh11	Original Macromedia Freehand-Dateien
*.jpg, *.png	Original Rasterbilder

Anhang C

Chronologische Liste der Änderungen

- 2002/01/07** `\newfloat{program}` repariert (auch ohne Chapter). Dank an Werner Bailer!
- 2002/03/06** Copyright-Notice an internat. Standard angepasst. Dank an Karin Kosina!
- 2002/07/28** „Studiengang“ → „Diplomstudiengang“
- 2003/08/24** Neues Macro: `\Messbox{breite}{hoehe}` – zur Kontrolle der Druckgröße ohne PS-Datei. Erweiterungen für Bakkalaureatsarbeiten
- 2005/04/09** Diverse Korrekturen: Captions von Tabellen nach oben gesetzt. `caption` auf neue Versionen adaptiert. `subfigure` wird nicht mehr verwendet
- 2006/01/20** Adaptiert zur Verwendung als Praktikumsbericht (2. Bakk.-Arbeit)
- 2006/03/24** Fehler in `\erklaerung` beseitigt (Dank an David Schwingenschlögl)
- 2006/04/06** Verwendung von T1-Fontencoding zur besseren Silbentrennung bei Umlauten etc.
- 2006/06/21** Neu: Bachelorstudiengang / Masterstudiengang. Literaturverweise auf Bakk-Arbeiten. `upquote.sty` eliminiert (Problem mit TS1-Kodierung). Verwende Komma (statt Punkt) als Trennzeichen in Dezimalzahlen.
- 2006/09/14** Anmerkungen zum Thema Plagiarismus.
- 2007/07/16** Ergänzungen für Code-Listings (`listings`) und Algorithmen (`algorithmicx`). BiBTeX-Datei aufgeräumt, Verwendung der Literaturformate verbessert. Komma (statt Punkt) als Trennzeichen in Dezimalzahlen wieder entfernt. Verwendung der `ae`-Fonts eliminiert (`cm-super` Fonts müssen installiert sein, ab MikTeX 2.5). Beispiel für Ersetzung in EPS-Dateien mit `psfrag`.

- 2007/10/04** Version 5.90: Das Laden der Pakete `inputenc` (Option `latin`) und `graphicx` (Option `dvips`) aus der Hauptdatei in die `sty`-Datei übertragen; `upquote` funktioniert nun. Paket `eurosym` ergänzt für Euro-Symbol (Anregung von Andreas Doubrava). Problem mit `colorpackage` repariert (gerasterter PDF-Ausdruck). Hinweise bzgl. Literatur ergänzt (`month`, `edition`), BibTeX-Datei gesäubert. Hinweis zum Einfügen von vertikalem Abstand zwischen Absätzen. Mathematik aufgeräumt, Verwendung von `amsmath`, Fallunterscheidungen. Diverse Änderungen bei Tabellen und Programmcode. Beispiele für BibTeX-Angaben von Spezialquellen: Audio-CDs, Videos, Filme. Einbinden von Dateien mit `\include{..}` Neue Datei: `_SimpleReport.tex` für kurze Reports (Projekte etc.).
- 2007/11/11** Version 5.91: Hinweise zur Einstellung der Output-Profile in TexNicCenter, Inverse Search Einstellung in YAP im Anhang.
- 2008/04/01** Version 6.00beta – kompletter Umbau! Auslagerung der Dokumenten-relevanten Teile in eine eigene `class`-Datei (`hgbthesis.cls`) mit Optionen. Die neue Style-Datei `hgb.sty` ist nun unabhängig vom Dokumententyp und nicht mehr kompatibel mit älteren Versionen! Die Liste der Änderungen ist jetzt in der Datei `_ChangeLog.tex` (DIESE Datei) und diese wird im Anhang eingebunden. Heading-Style auf Sans Serif geändert (ohne grausliche „Caps“).
- 2008/05/22** Neue Vorlage für Technical Reports (Klasse `hgbreport.cls`). Spracheinstellung nunmehr mit `babel`-Paket, Hauptsprache des Dokuments kann als Option der Klasse angegeben werden. Sprachumschaltung innerhalb des Dokuments funktioniert nun richtig. Mit der Sprachoption `german` wird automatisch die neue deutsche Orthographie (`ngerman`) verwendet. `babelbib` wird zur Formatierung des Literaturverzeichnis verwendet (neue BibTeX-Style-Optionen!). Header werden nunmehr mit `fancyhdr`-Paket erzeugt. Versionsnummerierung von `.cls` und `.sty` Files wird beendet (ab jetzt gilt: *Datum* = *Version*).
- 2008/06/10** Neues Listing-Environment `PhpCode`; bei allen Listing-Environments ist nun `mathescape=false` (kein Math-Mode nach \$). Bug bei Sprachumschaltung auf `ngerman` beseitigt.
- 2008/08/15** Diverse Kleinigkeiten in Literaturangaben überarbeitet (Dank an Norbert Wenzel), Spracheinstellung vereinheitlicht, Umlaute in `.bib`-Datei ersetzt.
- 2008/10/15** Zusätzliche Hinweise zur MikTeX-Installation (Windows) sowie LaTeX unter Mac OS X und Linux. Liste der Abkürzungen ergänzt.
- 2008/11/15** Diverse Schreibfehler korrigiert (Dank an Silvia Fuchshuber). Hinweis auf `sloppypar`-Umgebung.

- 2008/12/09** BibTeX-Tools: neuer Hinweis auf JabRef ergänzt, BibEdit entfernt (ist nicht mehr auffindbar).
- 2009/02/09** hgb.sty: Option „spaces“ zu url-Package ergänzt (ermöglicht gezielten Zeilenumbruch in URLs). Im allgemeinen Setup für listings: keepspaces=true; Obsoletes Environment sourcecode deaktiviert. Escape-Mode für LaTeXCode-Umgebung geändert. _DaBa.tex: Hinweis auf die Verwendung von \urldef für die Angabe von URLs in Captions. diplom (statt master) als Standard-Dokumententyp in _DaBa.tex („Diplomarbeit“). Neuer Abschnitt zum Umgang mit „Quellenangaben in Captions“. literatur.bib: alle URLs (bisher in note-Einträgen) auf url={..} geändert.
- 2009/04/14** Hinweis zum Einfügen einfacher Anführungszeichen ergänzt.
- 2009/07/18** Literaturangaben korrigiert und ergänzt.
- 2009/11/27** Experimentelle Version: Massive Änderungen, Umstieg auf pdflatex.
- 2010/06/15** Erstes Release der neuen Version mit pdflatex.
- 2010/06/23** Konflikt zwischen pdfsync-Package und array-Package (wird relativ häufig benutzt) durch \RequirePackage[novbox]{pdfsync} behoben. Seitenunterkante durch \flushbottom fixiert, variablen Absatzzwischenraum reduziert.
- 2010/07/27** Sprache der Erklärungsseite auf „german“ fixiert (auch wenn die Hauptsprache des Dokuments Englisch ist).
- 2010/12/03** Anmerkungen und Beispiele zum Zitieren von Gesetzestexten und Videos (Zeitangabe) ergänzt. Hinweis auf \nolinkurl{..} zur Angabe von Dateinamen.
- 2011/01/29** Einbau der Creative Commons Lizenz und entsprechender Hinweis in Abschnitt 3.6.1. Neue Makros \strictlicense, \cclicense und \license{...}. BibTeX-Einträge für Audio-CDs und Filme korrigiert, Beispiel für Online-Video ergänzt.
- 2011/02/01** Neues Makro \betreuerin{..} zur Angabe einer (weiblichen) Betreuerin.
- 2011/06/26** Umstellung der gesamten Literaturverwaltung auf biblatex mit dem Ziel, getrennte Abschnitte für verschiedene Kategorien von Einträgen im Quellenverzeichnis zu ermöglichen. Die Wahl fiel auf biblatex (es gäbe andere Optionen), weil damit BibTeX weiterhin nur einmal aufgerufen werden muss (und nicht für mehrere Dateien). Damit verbunden sind allerdings massive Änderungen bei der Syntax der BibTeX-Felder und es gibt auch mehrere neue Felder. Aktuell sind 3 Kategorien von Quellen vorgesehen, entsprechende Änderungen in hgbthesis.cls. Der klassische BibTeX-Workflow wird aktuell nicht mehr unterstützt, die Möglichkeit einer künftigen Dok-Option ist aber vorgesehen. Das Literatur-Kapitel ist komplett überarbeitet, die .bib-

Datei wurde ausgemistet. Neu ist die Empfehlung zur Aufnahme von Bildquellen in das Quellenverzeichnis, womit lange URLs in Captions (dort sind keine Fußnoten möglich) nicht mehr notwendig sind. „Persönliche Kommunikation“ als Literaturquelle entfernt (den Inhalt von Interviews sollte man direkt im Anhang wiedergeben). Das verwendete Bildmaterial wurde erneuert, aktuell werden nur mehr Public Domain Bilder verwendet. Das Kapitel „Hinweise für Word-Benutzer“ wurde endgültig entfernt. `\flushbottom` wieder auf `\raggedbottom` geändert, um übermäßige Abstände zwischen Absätzen zu vermeiden.

2012/05/10 Hinweis auf die in Österreich bislang nicht zulässige Verwendung von „Masterarbeit“ entfernt, `master` ist nunmehr die Default-Dokumentenoption. Anmerkungen zu lästigen `biblatex`-Warnungen ergänzt. Angaben für Windows-Programmpfade auf Win7 angepasst, MikTeX 2.9 als Minimalerfordernis.

Überflüssige Makros `\damonat` und `\dajahr` endgültig entfernt, statt `\abgabemonat` und `\abgabejahr` ist nun das neue Makro `\abgabedatum{yyyy}{mm}{dd}` vorgesehen (unter Verwendung von internen Zählern). Zur Formatierung von Datumsangaben wird das `datetime`-Paket verwendet.

Neue Fassung der eidesstattlichen Erklärung (inkl. englischer Version). PDF-Suche auf `synctex` umgestellt (`pdfsync`-Paket ist veraltet und wird nun nicht mehr verwendet).

Die älteren Dateiversionen von `algorithmicx.sty` und `algpseudocode.sty` (bisher explizit beigefügt) wurden weggelassen.

Hinweis auf die *Latin Modern Roman* OTF-Schriften ergänzt.

2012/07/21 Quellenverzeichnis: sprachabhängige Einstellung der Überschriften eingerichtet. Titel des Quellenverzeichnisses auf „Quellenverzeichnis“ (DE) bzw. „References“ (EN) fixiert. Makro `\MakeBibliography` hat damit keinen erforderlichen Parameter mehr.

2012/09/17 Wegen Änderungen im `biblatex`-package (Version 1.7, 2011/11/13) die Verwendung von BibTeX als backend eingestellt (`backend=bibtex8`).

2012/10/13 Option `lowtilde` beim URL-package eingestellt (erzeugt `~` statt `˜`).

2012/12/01 In Abschnitt 4.6.1 zusätzliche Code-Umgebungen ergänzt: `JsCode`, `PhpCode`, `HtmlCode`, `CssCode`, `XmlCode`.

2012/12/08 Die Code-Umgebungen in Abschn. 4.6.1 ergänzt und zur Verwendung von optionalen Argumenten erweitert (Hinweise in Abschnitt 4.6.1 auf die Argumente `firstnumber=last` und `numbers=none`). Quellenverzeichnis: den Eintragstyp `@software` für Games empfohlen und im Verzeichnis der Kategorie *avmedia* zugeordnet (Tab. 6.2 ergänzt). Game-Beispiel (von Manuel Wieser) und zusätzliche Tabelle 6.1 zur besseren Übersicht eingefügt.

2013/05/17 Wichtigste Änderung ist die vollständige Umstellung auf **UTF-8** unter Beibehaltung des **pdflatex**-Workflows. Damit sind zahlreiche weitere Modifikationen verbunden:

Alle Dateien (auch **.cls**, **.sty** und **.bib**) wurden auf UTF-8 konvertiert. Damit sollte es auch keine Probleme mehr mit Umlauten und Sonderzeichen unter MacOS geben.

Die verwendete Standard-Schriftfamilie ist nun „Latin Modern“ (**lmodern**). Sie ersetzt die „CM-Super“ Schriften, mit denen es immer wieder Installationsprobleme gab. Weiters wird jetzt das **cmap**-Paket zur besseren Such- und Kopierbarkeit von PDFs verwendet.

Das **listings**-Paket wurde durch **listingsutf8** ersetzt und für Umlaute im Quellcode adaptiert. Eventuell sind weitere Adaptierungen notwendig.

biber (min. Version 1.5!) wird nun anstatt **bibtex** (unterstützt keine UTF-8 Dateien) verwendet, zusammen mit **biblatex** (Version 2.5). Die Anweisung **\bibliography** wird (wieder) verwendet, allerdings nun in der Präambel, um die **.bib**-Datei im Fileverzeichnis anzugeben.

Das Makro **\C** (für die Menge der komplexen Zahlen \mathbb{C}) musste wegen Problemen in der T1-Kodierung ersetzt werden und heißt nun **\Cpx**. Die Makros **\R**, **\Z**, **\N**, **\Q** und **\Cpx** können nun auch außerhalb des Mathematik-Modus verwendet werden.

Der DVI-PS-PDF Workflow wird ab dieser Version überhaupt nicht mehr unterstützt. Damit ist auch das **psfrag**-Paket nicht mehr verwendbar. Entsprechende Hinweise wurden aus dem Text entfernt.

hyperref wurde auf UTF-8 umgestellt. Die grässlichen Standard-Rahmen und Farben der automatischen **hyperref**-Links wurden entfernt bzw. durch dezentere Farben ersetzt. Dadurch wird auch die Screen-Version der PDFs wieder lesbar.

Im Quellenverzeichnis wurde versuchsweise die **backref**-Option aktiviert. Damit werden bei allen Einträgen auch die zugehörigen Zitierstellen angegeben (erscheint durchaus sinnvoll).

Die bisherigen Korrekturen zur **biblatex**-Formatierung wurden entfernt, alles arbeitet nun mit Standard-Einstellungen. Die ursächlichen Probleme in **biblatex** scheinen in der aktuellen Version behoben zu sein.

Das Output-Profil für TeXnicCenter wurde für den neuen Workflow mit **biber** adaptiert und liegt nun in **_tc_output_profile_sumatra_utf8.tco**.

Das Windows-Script **_clean.bat** wurde entfernt, da TeXnicCenter nun ein eigenes „Clean Project“-Kommando aufweist (in „Build“).

Allgemeine Einstellungen zu *headings* und *biblatex* wurden aus der Datei **hgbthesis.cls** entfernt und in **hgbheadings.sty** bzw. **hgbbib.sty** verlagert. Diese können nun unabhängig verwendet wer-

den (s. Beispiel in `_TermReport.tex`).

Die Klassen-Datei `hgbtermreport.cls` wurde eliminiert, das Dokument `_TermReport.tex` basiert nunmehr auf der generischen LaTeX-Klasse `report` und verwendet keine eigene `.cls` Datei mehr.

2014/11/05 Neu: Logo auf der Frontseite bei allen Dokumententypen. Dazu gibt es ein neues Kommando `\logofile{pic}`, wobei `pic` der Name eine PDF-Datei im Verzeichnis `images/` ist. Falls *kein* Logo erwünscht ist, kann man die Zeile einfach weglassen oder durch `\logofile{}` ersetzen.

hyperref-Einstellungen: Einfärbung der Links wieder entfernt (`colorlinks = false`), weil beim Druck nicht abschaltbar. Stattdessen einheitliche (dezent) Rahmen für alle Linkarten. Zahlreiche Tippfehler eliminiert (Dank an Daniel Karzel).

Wegen eines Bugs in `biblatex 1.9` wurden die expliziten Abteilungen (`\-`) in `literatur.bib` vorübergehend entfernt (mit entsprechenden Folgen im Ergebnis). Der Bug soll in `biblatex 2.0` (derzeit noch nicht verfügbar) behoben sein.

Package `color` auf `xcolor` geändert. In `hgb.sty` neues „Convenience-Makro“ `\etc` ergänzt. Output-Profil für TeXnicCenter/SumatraPDF (Windows) repariert, forward/inverse Search funktioniert nun (Datei `_tc_output_profile_sumatra_utf8.tco`).

Anhang D

LaTeX-Quellcode

Hauptdatei _DaBa.tex

```
\documentclass[bachelor,german]{hgbthesis}
% Zulässige Class Options:
%   Typ der Arbeit: diplom, master (default), bachelor, praktikum
%   Hauptsprache: german (default), english
%%-----

\graphicspath{{images/}} % name of directory containing the images
\logfile{FH00eIntl2014} % name of PDF, remove or use \logfile{} for no logo
\bibliography{literatur} % name of the BibTeX (.bib) file

%%-----
\begin{document}
%%-----

% Einträge für ALLE Arbeiten: -----
\title{Konzeption eines Mail Service}
\author{Ing. Thomas Herzog}
\studiengang{Software Engineering}
\studienort{Hagenberg}
\abgabedatum{2016}{02}{01} % {YYYY}{MM}{DD}

%% zusätzlich für eine Bachelorarbeit: -----
\nummer{S1310307011-A} % XX...X = Stud-ID, z.B. 0310238045-A
% (A = 1. Bachelorarbeit)
\semester{Sommersemester 2016}
\gegenstand{Software Engineering}
\betreuer{Dipl. Ing. Dobler Heinz} % oder \betreuerin{..}

%% zusätzlich für einen Praktikumsbericht: -----
\nummer{XXXXXXXXXX-B} % XX...X = Stud-ID, z.B. 0310238045-B
% (B = 2. Bachelorarbeit)
\betreuer{Mag.-Pjotr I.-Czar\Creative Director} % \betreuerin{..}
\firma{Oligarchic Media International GmbH\}
```

```

    Online Division\\
    1020 Wien, Hubertusgasse 3a
}
\firmenTel{1-234-5678-100}
\firmenUrl{www.mogul.at}

%\strictlicense % erzeugt restriktive Lizenzformel

%%%------
\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{schluss}

%%%------
%%%-Anhang
\appendix
\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

%%%------
\MakeBibliography
%%%------

%%%-Messbox zur Druckkontrolle
\include{messbox}

\end{document}

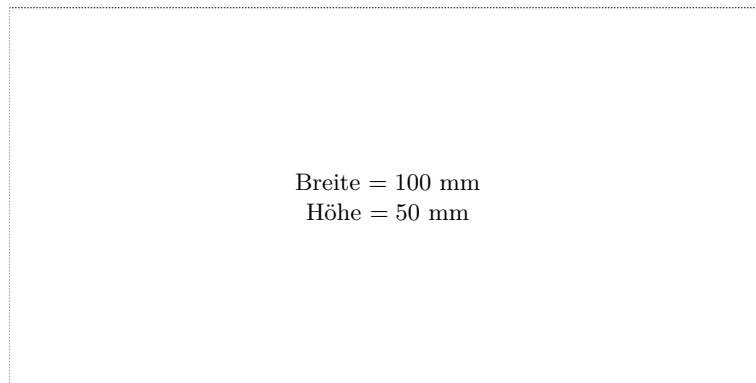
```

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

Quellenverzeichnis

Messbox zur Druckkontrolle

— Druckgröße kontrollieren! —



— Diese Seite nach dem Druck entfernen! —