

Skriptum Wissenschaftliche Arbeiten schreiben mit \LaTeX

814014 VU2

Petra Seibert
BOKU Wien

Version 28. Mai 2021



CTAN lion drawing by Duane Bibby

© Petra Seibert (BOKU Wien), 2016-2021. Frei zum persönlichen Gebrauch.
Für andere Verwendung: Bitte um Rücksprache.
Meldungen von Fehlern und Verbesserungsvorschläge werden gerne entgegen genommen!

Inhaltsverzeichnis

1.	Installation	5
1.1.	Voraussetzung	5
1.2.	Windows	5
1.3.	MacOS	5
1.4.	Linux	6
1.5.	TeXstudio	6
2.	Allgemeines zu L^AT_EX	7
2.1.	WYSIWYG (Word & Co.), L ^A T _E X, T _E X	7
2.2.	Hilfen und Literatur	7
3.	Erste Schritte	9
3.1.	Ablauf	9
3.2.	Aufbau eines L ^A T _E X-Quelldokuments	11
3.3.	Dokumentklassen	11
3.4.	Verwendbare Zeichen	11
4.	Umbruch und Gliederung, Fußnoten und Randnotizen	12
4.1.	Zeilen- und Seitenumbruch	12
4.2.	Gliederung	12
4.3.	Fußnoten und Randnotizen	13
5.	Zeichensätze und Sonderzeichen	14
5.1.	Zeichensätze und Codetabellen	14
5.2.	Sonderzeichen	15
6.	Längenangaben und Abstände	17
6.1.	Längeneinheiten	17
6.2.	Einfügen von Abständen	17
7.	Schriftformatierung	18
7.1.	Schriftgröße	18
7.2.	Weitere Schriftmerkmale	18
7.3.	Schriftarten (Fonts)	19
7.4.	Typografische Feinheiten	21
7.5.	Spezielle Schriftauszeichnungen	21
7.6.	Zusammenfassung: Wie gehe ich vor, um die Schriftart und -formatierung zu ändern?	22
8.	Boxen	23
8.1.	Das Box-Konzept in L ^A T _E X	23
8.2.	Links-Rechts-Box (LR-Box)	23
8.3.	Absatzboxen (parbox)	24
8.4.	Besondere Boxen	25
9.	Texte in anderen Sprachen als Englisch	26
9.1.	Was ist sprachspezifisch in L ^A T _E X?	26
9.2.	Sprachen wählen	26
9.3.	Eingabe von (deutschen) Sonderzeichen	27
10.	Absatzformatierung	28
10.1.	Einzug und Absatzabstand ändern	28
10.2.	Ausrichtung ändern	28

10.3.	Umgebungen für Code und dergleichen	29
10.4.	Zeilenabstand ändern	29
10.5.	Silbentrennung	30
11.	Neu- und Umdefinition von Befehlen und Umgebungen	31
12.	Aufteilung eines Dokuments auf mehrere Dateien	32
12.1.	<code>\input</code>	32
12.2.	<code>\include</code>	32
13.	Listen	33
13.1.	Eingebaute Listen	33
13.2.	Pakete zur Listengestaltung	34
14.	Tabellen	37
14.1.	Einfache Tabelle	37
14.2.	Zellen, die über mehrere Spalten oder Zeilen gehen	38
14.3.	Pakete zur Verbesserung und Erweiterung des Tabellen-Layouts	38
14.4.	Einbringen von Werten in eine tabular-Umgebung	42
15.	Abbildungen	45
15.1.	Allgemeines zu Grafikdateiformaten	45
15.2.	PS- und PDF-Format.	47
15.3.	Erzeugen von Vektor-(Postscript-)Grafiken	48
15.4.	Umwandeln von Grafikdateien	49
15.5.	Einbinden von Grafiken in Latex	50
15.6.	Arbeiten mit <code>tikz</code>	52
15.7.	Zusammenfassende Empfehlungen	52
16.	Gleitumgebungen (floats)	54
16.1.	Gleitende Tabellen und Abbildungen mit Legende	54
16.2.	Wichtiges Zusatzpaket: <code>caption</code>	56
16.3.	Weitere Zusatzpakete	56
17.	Literaturverweise und -verzeichnis	59
17.1.	Einführung	59
17.2.	Literatur mit <code>bibtex</code> und <code>natbib</code>	60
17.3.	Konvertieren von Literaturdatenbanken	64
18.	Seitenlayout	66
18.1.	Papierformat und Ränder mit <code>geometry</code>	66
18.2.	Kopf- und Fußzeilen, <code>fancyhdr</code>	68
18.3.	Mehrspaltiger Satz mit <code>multicol</code>	70
18.4.	Layoutempfehlungen	71
18.5.	Formatierung von Kapitelüberschriften und Verzeichnissen	71
18.6.	Alternative Pakete	72
19.	Satz mathematischer Formeln	73
19.1.	Arten von mathematischen Umgebungen	73
19.2.	Satzregeln und Satz-Beispiele	73
19.3.	Fonts mit mathematischen Zeichen	74
19.4.	AMSMath und andere Zusatzpakete	75
20.	Satz von physikalischen Einheiten mit <code>SIunitx</code>	77
20.1.	Grundsätzliches zum Setzen von Einheiten.	77
20.2.	Verwendung von <code>siunitx</code>	77

21.	Satz von chemischen Formeln	79
21.1.	Summenformeln mit mhchem	79
21.2.	Sicherheitshinweise mit hpstatement und rsphrase	79
21.3.	Strukturformeln mit chemfig	80
22.	Farben in L^AT_EX	82
22.1.	Farbpakete	82
22.2.	Farbmodelle	82
22.3.	Farben definieren	83
22.4.	Verwendung von Farben	83
23.	Erzeugen und Verarbeiten von PDFs	84
23.1.	Versionen des PDF-Standards	84
23.2.	Einbindung von Seiten aus anderen Dokumenten als eigene Seite	85
23.3.	Erstellung von Broschüren mit zwei Textseiten pro Druckseite	86
23.4.	Weitere PDF-Tools	86
24.	Export / Import (html, doc, rtf, etc.)	87
24.1.	Office-Dokumente nach L ^A T _E X exportieren	87
24.2.	L ^A T _E X in HTML-, ePUB-, Office- (doc[x], odt) Format exportieren	87
25.	Liste von weiteren Paketen und Tools	88
26.	Präsentationen und Poster mit beamer	90
26.1.	Paket beamer	90
26.2.	beamerposter	92
A.	PDF-Handling aus dem IMG-Wiki	94

1. Installation

TeXlive bzw. MacTeX und das *Integrated Development Environment* TeXstudio installieren. Es gibt eine detaillierte Anleitung für die TeXlive-Installation unter Berücksichtigung aller Betriebssysteme bei der englischsprachigen TUG (TeX User Group), auch auf Deutsch, siehe <http://tug.org/texlive/doc/texlive-de/texlive-de.html> (englische Version: <http://tug.org/texlive/doc/texlive-en/texlive-en.html>). Dabei wird (ausgenommen bei Installation als Linux-Paket) auch der TeXlive-Manager tlmgr mitinstalliert, den man verwenden kann, um Updates einzuspielen oder einzelne Pakete nachzuladen (falls man keine vollständige Installation gemacht hat). Man kann natürlich auch durch Neuinstallation aktualisieren.

Es gibt noch weitere Softwarepakete, die beim Arbeiten mit L^AT_EX nützlich sind, insbesondere ghostscript und ghostview bzw. pdfbox, sowie jabref.

1.1 Voraussetzung

Ca. 4 GB freier Speicherplatz (je nachdem, wie vollständig die Installation sein soll – evt. auch mehr) und halbwegs schnelle Internetverbindung (alternativ Installation via DVD). Es geht nicht in 5 min, also ein wenig Zeit dafür reservieren.

1.2 Windows

Eine Kurzbeschreibung der Installation über den Online-Installer gibt es unter <https://www.dante.de/installation-von-tex-live-unter-windows/>.

Neben der Basisinstallation gibt es auch die “advanced“-Version; diese bietet während des Installationsvorgangs zusätzliche Abfragen an. Relevant ist im wesentlichen, dass man Sprachen abwählen kann, die man sicher nie braucht, um so Platz zu sparen. Es schadet aber nicht, diese mitzuinstallieren, wie dies in der einfachen t1-install-Version passiert.

[Optional, aber empfohlen:] *Ghostscript und Ghostview installieren* (dient uns dazu, in PDF- und PS-Files Ausschnitte festzulegen):

Ghostscript ist in texlive enthalten. Von ghostview scheint es derzeit leider keine Windows10-taugliche Version zu geben. Das Paket *pdfbox* ist ein guter Ersatz!

Die L^AT_EX-Entwicklungsumgebung mit GUI TeXstudio installieren: Von <http://texstudio.sourceforge.net/> den Windows-Installer herunterladen und ausführen.

1.3 MacOS

Die TeXlive-Distribution für MacOS ist MacTeX (<http://www.tug.org/mactex/>). Es ist von <http://www.tug.org/mactex/mactex-download.html> herunterladbar. Dort findet sich auch eine kleine Installationsanleitung.

[Optional, aber empfohlen:] *Ghostscript und Ghostview installieren* (dient uns dazu, in PDF- und PS-Files Ausschnitte festzulegen): Sollte bei MacTeX bereits inkludiert sein!

Die L^AT_EX-Entwicklungsumgebung mit GUI TeXstudio installieren: Von <http://texstudio.sourceforge.net/> die OSX-Version herunterladen und installieren.

1.4 Linux

TeXlive kann unter Linux sowohl als Paket der jeweiligen Distribution als auch direkt mit Hilfe des TeXlive managers (tlmgr) installiert werden. Dann hat man aber bei anderen Paketen wie z.B. TeXstudio Paketabhängigkeiten, die nicht erfüllt sind. Daher empfehle ich, obwohl in der Regel nicht ganz so aktuell, die Pakete der Distribution zu installieren.

Für Fortgeschrittene gibt es unter <https://www.tug.org/texlive/debian.html>, Section *Integrating vanilla T_EX Live with Debian*, eine Anleitung, wie man das aktuelle TeXLive-System in ein Debian-System integrieren kann. Sollte auch für Debian-Derivate wie Ubuntu oder Mint funktionieren.

Es wird davon ausgegangen, das Linux-NutzerInnen wissen, wie man Pakete installiert. Bei vielen Distributionen ist TeXlive auch schon vorinstalliert, zumindest die Basispakete. Unter Ubuntu gibt es ein meta-Paket für die volle Installation namens `TeXlive-full`. Daraus ergeben sich die Schritte

- ▶ Prüfen, ob `texlive` schon auf dem System vorhanden ist.
 - Wenn ja, ggf. gewünschte Ergänzungspakete (alle Pakete mit `tex` im Namen anzeigen lassen) nachinstallieren.
 - Wenn nein, am einfachsten `texlive-full` (oder äquivalent) installieren.
- ▶ Ghostview und ghostscript sind normalerweise auf jedem Linux-System vorhanden, Pakete heißen `gv` und `ghostscript`.
- ▶ Das Paket `texstudio` installieren.

1.5 TeXstudio

Anschließend kann TeXstudio gestartet und unter Options / TeXstudio konfiguriert werden (unter Umständen hat das Programm nicht alle Programme wie `latex.exe`, `pdflatex.exe` etc. gefunden, dann kann man sie hier konfigurieren). Die Konfigurationsmöglichkeiten sind sehr umfangreich, man sollte sie in Ruhe studieren! Dabei sollte man gleichauf der ersten Konfigurationsseite ganz unten *Show Advanced Options* aktivieren, damit man alle sieht.

2. Allgemeines zu L^AT_EX

2.1 WYSIWYG (Word & Co.), L^AT_EX, T_EX

Unterschiede, Vor- und Nachteile

T_EX, L^AT_EX kurze Geschichte

Nach L^AT_EX 2_ε bzw. pdfL^AT_EX: L^AT_EX 3.0, Omega, LuaT_EX, XeL^AT_EX

(wird in der Lehrveranstaltung mündlich besprochen; siehe auch englischsprachige Kurzbeschreibung¹)

Not So Short Intro
p. 1–4

2.2 Hilfen und Literatur

- ▶ DANTE, Deutschsprachige Anwendervereinigung TeX e.V.
<https://www.dante.de/>
 - <https://texdoc.net/> Alle Paketdokumentationen online
 - <https://texfragen.de/> Neue FAQ (deutschsprachig)
 - <https://texwelt.de/> Forum, deutschsprachig
 - <https://golatex.de/> Noch ein Forum, auch deutschsprachig
 - <https://tex-talk.net/> Blog
 - <https://texample.net/community/> T_EX-blog-aggregator
 - <https://latex.org/forum/> Gutes Forum, kommt leider in Suchmaschinen selten heraus, daher lieber bookmarken und direkt dort suchen (oder bei der Suche `site:latex.org` dazu geben).
- ▶ Suchmaschinen zeigen meist <https://tex.stackexchange.com/>.
- ▶ UK TeX User Group <http://www.tug.org/>
 - <https://texfaq.org/>
 - Unter <http://tex.world/list/> gibt's zum Glück noch eine Übersicht der DANTE/TUG-nahen Webseiten.
- ▶ *L^AT_EX 2_ε Kurzbeschreibung* haben hoffentlich schon alle.
Englische Version: <http://www.ctan.org/tex-archive/info/lshort/english>;
deutsche Version (weniger ausführlich!): <https://ctan.org/texarchive/info/german/LaTeX2e-Kurzbeschreibung>. Verweise in diesem Skript: „Kurzbezeichnung“ bezieht sich auf die deutsche Version, „Not So Short Intro“ auf die englische.
- ▶ Wikibook
 - L^AT_EX-Kompodium (<https://de.wikibooks.org/wiki/LaTeX-Kompodium>)
 - L^AT_EX Wikibook English (<https://en.wikibooks.org/wiki/LaTeX>)

¹In *Randnotizen* wird auf die entsprechenden Seiten der deutsch- bzw. englischsprachigen, quasi-offiziellen „Kurzbeschreibung“ verwiesen. Diese sollten auch tatsächlich herangezogen werden, da sie oft zusätzliche Informationen enthalten, die nicht in diesem Skriptum wiederholt werden.

- ▶ Herbert Voß: Einführung in L^AT_EX – Unter Berücksichtigung von pdfLaTeX, XeLaTeX, LuaLaTeX. DANTE e.V., Lehmanns media, 2017 (3. Aufl.), 960 S. ISBN 978-3-86541-798-5. 29,95 € bzw. 25,- € für Mitglieder von DANTE e.V. sowie die weiteren Bücher der DANTE-Edition (<https://www.dante.de/dante-e-v/literatur/>)
- ▶ Helmut Kopka: *L^AT_EX – Band 1, Einführung*. Pearson Education, 3. Auflage, 2002. 521 Seiten, CD-ROM. ISBN 978-3-86894-088-6; 39,95 €
- ▶ Frank Mittelbach, Michel Goossens: *Der L^AT_EX-Begleiter*. Pearson Studium, 2010. 1168 Seiten, CD-ROM. ISBN 3-8273-7166-X; 39,95 €. Vergriffen!
- ▶ dasselbe in Englisch: *The L^AT_EX Companion*, Addison-Wesley Publ., 2004. 1090 Seiten, CD-ROM, ISBN 0-201-36299-6; 69,95 €. Siehe dafür und für weitere Bücher: <https://www.latex-project.org/help/books/>.

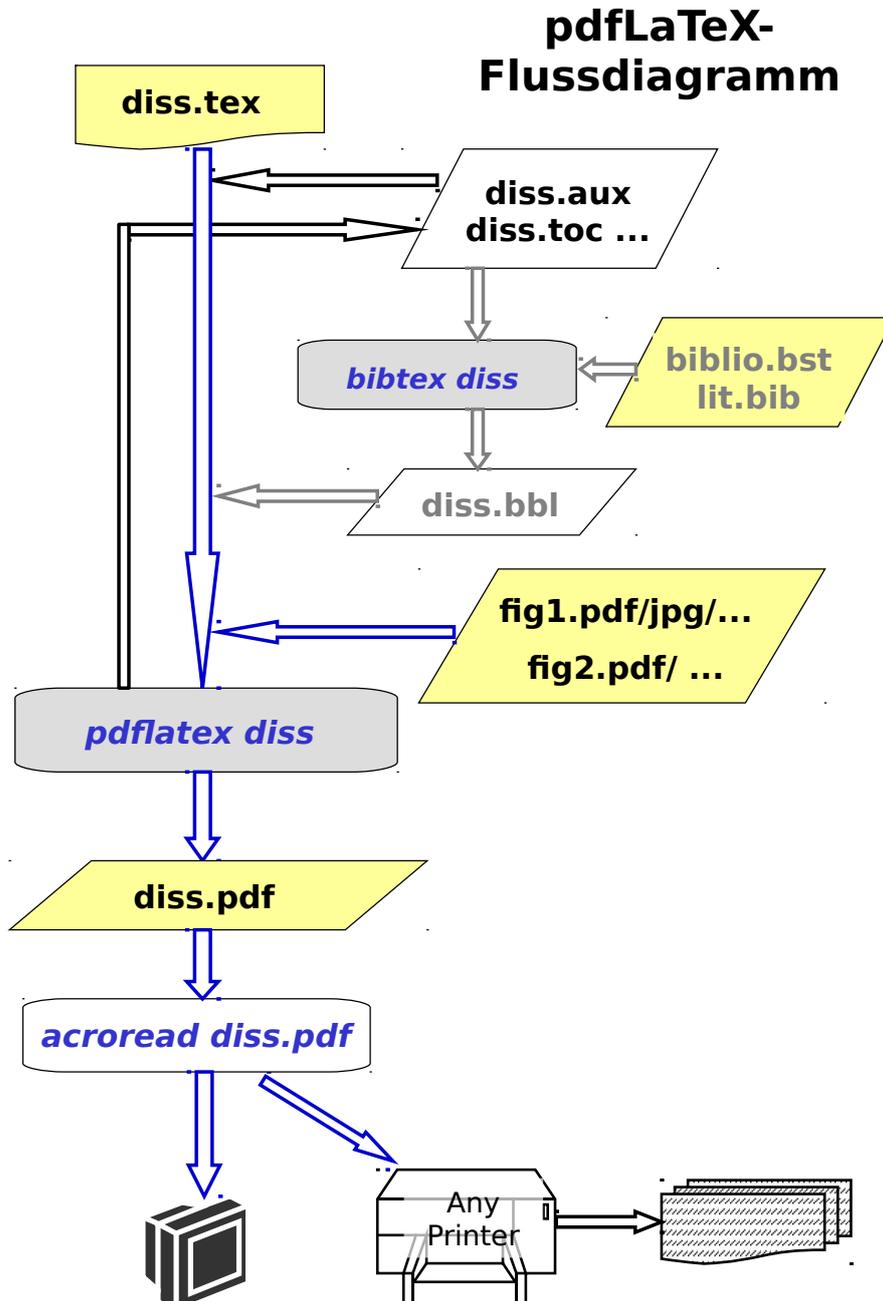
Für Bücher siehe auch Verzeichnis der Versandbuchhandlung Lehmanns:
<http://www.lehmanns.de/fachgebiet/45-mathematik-informatik/10-informatik/90-weitere-themen/40-latex>

3. Erste Schritte

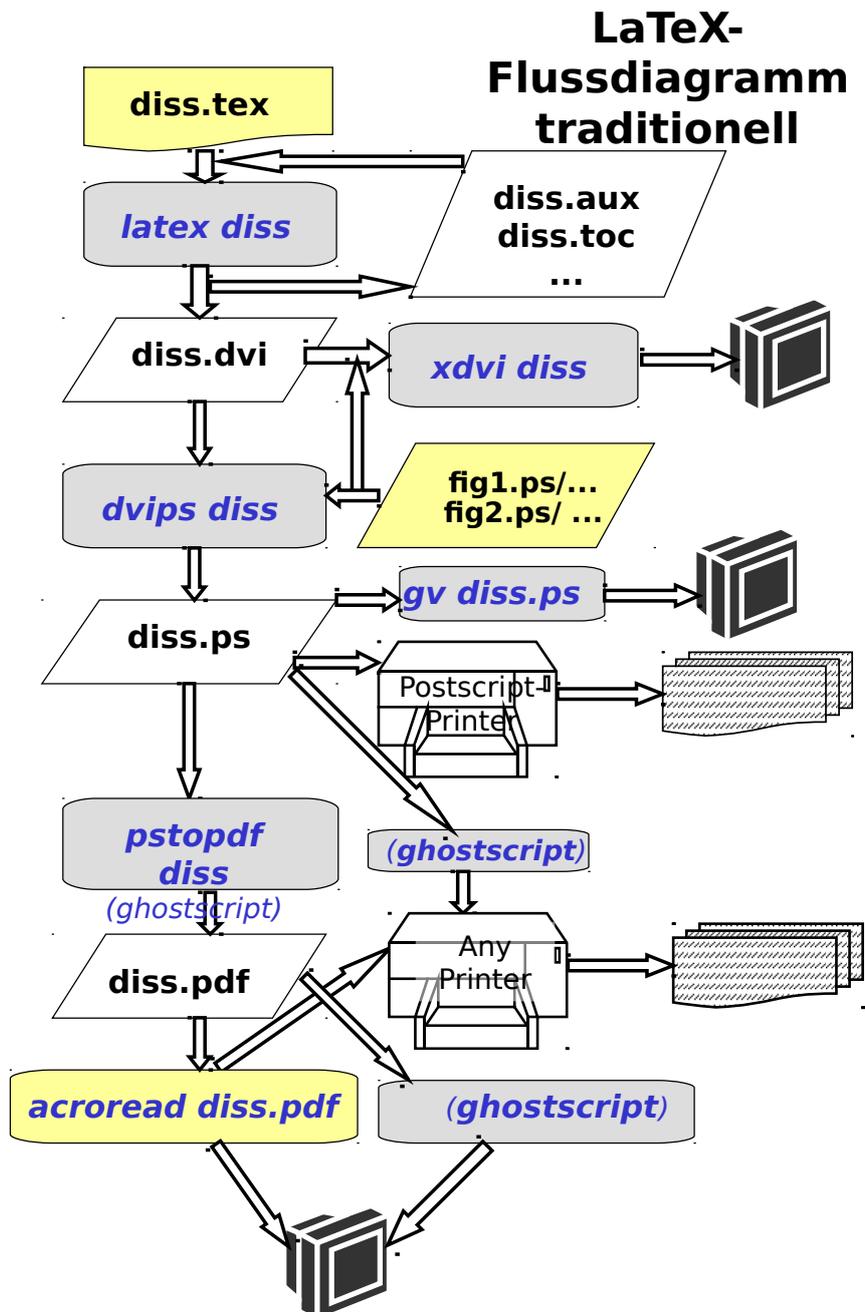
3.1 Ablauf

Details siehe Kurzbeschreibung

Kurzbeschreibung
S. 7-8



Die „traditionelle“ Version (mit latex statt pdflatex als *engine*) produziert ein Zwischenformat .dvi (device-independent), das noch keine Grafikfiles enthält:



3.2 Aufbau eines L^AT_EX-Quelldokuments

(.tex-File):

```
%preamble (header): [everything after % is a comment]
\documentclass[options]{classname}
...
\usepackage[options]{package}
%Inhalt:
\begin{document}
  content...
\end{document}
Alles was hier kommt wird ignoriert.
```

Kurzbeschreibung
S. 10-11, 13

3.3 Dokumentklassen

Details siehe Kurzbeschreibung

Kurzbeschreibung
S. 11-13

3.4 Verwendbare Zeichen

- ▶ Bedeutung von Leerzeichen (<tab> <cr>), Absätzen (<cr><cr>), \\
- ▶ Zeichen mit Sonderbedeutung

Bei jedem L^AT_EX-System dürfen mindestens die folgenden Zeichen zur Eingabe von Text verwendet werden:

a...z A...Z 0...9 . : ; , ? ! ' ' () [] - / * @ + =

Die folgenden Eingabezeichen haben für L^AT_EX eine Spezialbedeutung oder sind nur innerhalb von mathematischen Formeln erlaubt:

\$ & % # _ { } ~ ^ " \ | < >

Zum Beispiel markiert \ einen Befehl: \LaTeX gibt L^AT_EX aus.

Mehr zu Sonderzeichen und Symbolen später.

Kurzbeschreibung
S. 6

Kurzbeschreibung
S. 14-15, 21-22

4. Umbruch und Gliederung, Fußnoten und Randnotizen

4.1 Zeilen- und Seitenumbruch

Kurzbeschreibung
S. 16-17

	<i>Zeile</i>	<i>Seite</i>
„hart“	<code>\\</code> oder <code>\newline</code>	<code>\newpage</code>
„weich“	<code>\linebreak[number]</code>	<code>\pagebreak[number]</code>

„weich“: die Zeile bzw. Seite wird bei Blocksatz bis zum Rand aufgefüllt
`number` 1 . . . 4 (4 = verpflichtend), keine Angabe =4

Optionale Modifikationen von `\\`:

`*` Seitenumbruch mit diesem Zeilenumbruch **nicht** erlaubt!

Zu vermeiden: Zeile die mit `\\` endet, danach Leerzeile: bedeutet in der Ausgabe leere Zeile + neuer Absatz. Am Ende eines Absatzes hat leere Zeile keinen Sinn. Absatzabstände mit entsprechenden Kommandos oder Parametern, siehe später!

4.2 Gliederung

Kurzbeschreibung
S. 22-23

`\maketitle` erzeugt einen **Titel** aus `\author{names}`, `\title{text}` und `\date{text}`, die im Header definiert werden (wenn `date` nicht definiert, wird aktuelles Datum eingesetzt). Bei `article` auf der Seite, bei `report` und `book` eigene Seite (kann mit Dokumentklassenoption `titlepage/notitlepage` angepasst werden).

Gliederungsebenen – siehe TeXstudio. Man sollte Subebenen nicht verwenden, ohne dass vorher die höchste Ebene (`article`: `section`, `report / book`: `chapter`) verwendet wurde, sonst lautet die Nummerierung z. B. [0.1 Überschrift](#).

Syntax: `\section[short]{title}`

Inhaltsverzeichnis ausgeben: `\tableofcontents`

Tabellen- und Abbildungsverzeichnis: `\listoftables`, `\listoffigures`

`\section*[short]{title}` – nicht nummeriert und nicht im Inhaltsverzeichnis. Doch ins Inhaltsverzeichnis bringen:

`\addcontentsline{file}{secunit}{entry}`

`\phantomsection` (wenn `hyperref`-Paket verwendet wird, für korrekten Link)

wobei

`file` `toc`

`secunit` `section, subsection, etc.`

`entry` `(short) title`

4.3 Fußnoten und Randnotizen

Fußnoten

`text1 text\footnote{Der Fußnotentext.}`
(ohne Leerzeichen zwischen Text und Fußnote[nzeichen]!)

Kurzbeschreibung
S. 23

Randnotiz

`\marginpar[textleft]{textright}`

`textright`

Im Falle eines doppelseitigen Layouts (`twoside`, siehe später) erscheint die Randnotiz auf geraden Seiten am linken Rand. Um zum Beispiel die Ausrichtung außen und innen spiegelverkehrt zu ermöglichen, wird in diesem Fall der Inhalt des optionalen Parameters (im Beispiel `textleft`) angezeigt, sofern vorhanden.

Bei doppelseitigem Layout mit Randnotizen sollte auch

`\usepackage{mparhack}`

verwendet werden, um einen unter gewissen Umständen auftretenden Bug von T_EX bei der Links-Rechts-Zuordnung zu beheben.

¹Der Fußnotentext.

5. Zeichensätze und Sonderzeichen

5.1 Zeichensätze und Codetabellen

Siehe auch http://en.wikipedia.org/wiki/Code_page

128 Zeichen ASCII Zeichensatz (7 bit *character set*), enthält Steuerzeichen (bel= bell, cr = carriage return, nl = new line, . . .), Satz- und Sonderzeichen (nicht alle) sowie Klein- und Großbuchstaben des englischen Alphabets und die Ziffern.

0 nul	1 soh	2 stx	3 etx	4 eot	5 enq	6 ack	7 bel	
8 bs	9 ht	10 nl	11 vt	12 np	13 cr	14 so	15 si	
16 dle	17 dc1	18 dc2	19 dc3	20 dc4	21 nak	22 syn	23 etb	
24 can	25 em	26 sub	27 esc	28 fs	29 gs	30 rs	31 us	
32 sp	33 !	34 "	35 @	36 \$	37 %	38 &	39 '	
40 (41)	42 *	43 +	44 ,	45 -	46	47 /	
48 0	49 1	50 2	51 3	52 4	53 5	54 6	55 7	
56 8	57 9	58 :	59 ;	60 <	61 =	62 >	63 ?	
64 @	65 A	66 B	67 C	68 D	69 E	70 F	71 G	
72 H	73 I	74 J	75 K	76 L	77 M	78 N	79 O	
80 P	81 Q	82 R	83 S	84 T	85 U	86 V	87 W	
88 X	89 Y	90 Z	91 [92 \	93]	94 ^	95 _	
96 '	97 a	98 b	99 c	100 d	101 e	102 f	103 g	
104 h	105 i	106 j	107 k	108 l	109 m	110 n	111 o	
112 p	113 q	114 r	115 s	116 t	117 u	118 v	119 w	
120 x	121 y	122 z	123 {	124	125 }	126 ~	127 del	

Betriebssysteme arbeiten in der Regel mit 8-bit character sets. Damit sind diverse Sonderzeichen (z. B. mathematische) und zusätzliche nationale Zeichen darstellbar. Während die unteren 128 Zeichen immer gleich sind, gibt es verschiedene Varianten von den oberen, um verschiedenen Sprachen (nordische, slawische, usw.) gerecht werden zu können. Diese Varianten heißen *code pages*. Historisch wurden in verschiedenen Betriebssystemen selbst in der gleichen Sprachregion unterschiedliche Codiertabellen verwendet, z. B.

- ▶ latin-1 in Linux <https://en.wikipedia.org/wiki/Latin-1>
- ▶ cp1252 in Windows <https://en.wikipedia.org/wiki/Windows-1252>
- ▶ MacRoman in MacOS https://en.wikipedia.org/wiki/Mac_OS_Roman

mit dem Resultat von „Zeichensalat“ beim Austausch von Files.

Deshalb wurde mit *Unicode* eine Vereinheitlichung geschaffen. Es gibt verschieden Unicode-Definitionen, UTF-8 (auch *utf-8*, *utf8*) hat sich durchgesetzt. Ein Zeichen wird hier mit 1 bis 4 Bytes (à 8 bit) dargestellt. ASCII ist eine Untermenge. Es können >1.000.000 Zeichen dargestellt werden, das reicht für die meisten Sprachen auf der Welt.

Die Codierung ist offensichtlich bei der Eingabe und bei der Ausgabe relevant.

5.1.1 Eingabecodierung

Seit L^AT_EX-2018 wird utf8-Kodierung automatisch angenommen, (Mit `\UseRawInputEncoding` in der Präambel kann man auf 7-bit ASCII zurückschalten.)

Bei älteren Versionen kann man Paket `inputenc` mit der Option `utf8` laden um z. B. Umlaute direkt eingeben zu können:

```
\usepackage[utf8]{inputenc}
```

(`utf8x` normalerweise nicht verwenden!)

```
\usepackage[utf8]{inputenx}
```

(mehr Zeichen enthalten, erlaubt Umschalten im Dokument)

Die Eingabecodierung hat nur dann das gewünschte Ergebnis, wenn das Zeichen auch im Font für die Ausgabe darstellbar ist!

5.1.2 Ausgabecodierung

```
\usepackage[T1]{fontenc}
```

T1 steht für Type 1 und ist unbedingt erforderlich, damit Texte mit Umlauten und anderen Akzenten korrekt umgesetzt werden (der Unterschied ist beim Hinschauen nicht zu erkennen, aber relevant z. B. für Silbentrennung, Umwandeln von PDF in Text, Suchen von Wörtern mit Umlaut, usw.)

OT1 ist die old T_EX version, sollte nicht mehr verwendet werden (aber default wenn Paket nicht geladen wird!). Andere Optionen sind für westeuropäische Sprachen nicht relevant.

5.2 Sonderzeichen

1. „Verbotene“ Zeichen: manche können durch escapen mit `\` dargestellt werden. (Kurzbeschreibung S. 21)
2. Verbatim: `\verb|...|`
3. Paket `textcomp`. Nicht mit allen Schriftarten kompatibel.
4. Paket `pi font` – Access to PostScript standard Symbol and Dingbats fonts
5. Weitere Pakete wie `gensym`, `marvosym`, `wasysym`
6. L^AT_EX-spezifische Syntax für
 - Umlaute, Akzente, etc. (Kurzbeschreibung S. 22)
 - typografische Anführungszeichen
 - typografische Gedankenstriche
 - mathematischer Modus

Wichtige Beispiele:

- typografische Anführungszeichen (für Englisch) `"..."` `"\ldots"`

`"X"` `"X"` einfache: `'X'` `'X'`

`“` `”`

• • • in Computer Modern (die T_EX-Schrift von D. Knuth)

- Größer-/Kleiner-Zeichen `<>` `$<>$`, `<>` `\textless \textgreater`
- Gradzeichen `°` `$^\circ$`, `°` `\textdegree`
- – (n-dash, Gedankenstrich) `--`
- — (m-dash, Geviertstrich, amerikanischer Gedankenstrich) `---`
- Paragraph `§` `\S`

- Euro-Symbol € (`\texteuro` aus paket `textcomp`), oder € (`\euro` aus Paket `eurosym`) verwenden. Mit den Paketen `textcomp` und `(utf8)-inputenc` kann das €-Zeichen auch über das Keyboard eingegeben werden (de: `⌘-grzE`)

Achtung, der Leerraum `␣` nach `\irgendetwas␣` wird aus syntaktischen Gründen verschluckt. Abhilfe (hier am Beispiel Euro): `\euro{}␣AAA` oder `\euro\␣AAA` oder `\euro~AAA` (letzteres ist ein nicht dehn- und umbrechbarer Zwischenraum!)

Das Logo L^AT_EX kann man mit `\LaTeX` erzeugen.

Die Referenz: *The Comprehensive L^AT_EX Symbol List*,

<http://mirror.easyname.at/ctan/info/symbols/comprehensive/symbols-a4.pdf>

6. Längenangaben und Abstände

6.1 Längeneinheiten

Kurzbeschreibung
S. 46–48
Not So Short Intro
p. 113–114

Mögliche Einheiten für Längen (Zahlenwerte können auch negativ sein!):

mm	Millimeter
cm	Zentimeter = 10 mm
in	inch = 25.4 mm
pt	point = $\frac{1}{72.27}$ in = $\frac{800}{803}$ bp ≈ 0.351 mm (American point, T _E X point)
bp	big point = $\frac{1}{72}$ in = 0.352778 mm ≈ 0.353 mm (DTP point)
em	Geviert (doppelte Breite einer Ziffer der aktuellen Schrift)
ex	Höhe des Buchstabens x der aktuellen Schrift

Einige „exotischere“ Einheiten findet man in der *L^AT_EX Referenz der Umgebungen, Makros, Längen und Zähler* (PDF, siehe Moodle).

6.2 Einfügen von Abständen

Horizontal: `\hspace{length}` `\hspace*{length}` `\hskipLENGTH`

Vertikal: `\vspace{length}` `\vspace*{length}` `\vskipLENGTH`

Beispiel: `\hspace{1em}`, `\vspace{1cm}`, `\vskip1cm`

Zwischenraum, der mit der *-Form eingefügt wurde, kann nicht verschluckt werden. Die skip-Befehle sind T_EX, nicht L^AT_EX.

Für beide Befehle gibt es eine erweiterte Syntax für die Längenangabe length:

`<dimen1> plus <dimen2> minus <dimen3>`

mit der elastische Abstände durch sog. *glue* erzeugt werden. Die zusätzlichen Argumente sind optional. (Im Math-Mode gibt es noch ein `\mskip`.)

Vordefinierte Befehle für horizontale Abstände:

<code>\,</code>	ein sehr kleiner Abstand: ■■■
<code>\enspace</code>	so breit wie eine Ziffer: ■ ■
<code>\quad</code>	so breit, wie ein Buchstabe hoch ist („weißes Quadrat“): ■ ■
<code>\qqquad</code>	doppelt so breit wie ein <code>\quad</code> : ■ ■
<code>\hfill</code>	ein Abstand, der sich von 0 bis ∞ ausdehnen kann. ■ ■

Vordefinierte Befehle für vertikale Abstände:

(Wirken nur *zwischen* Absätzen!)

<code>\smallskip</code>	etwa 1/4-Zeile
<code>\medskip</code>	etwa 1/2-Zeile
<code>\bigskip</code>	etwa 1 Zeile
<code>\vfill</code>	ein Abstand, der sich von 0 bis ∞ ausdehnen kann

Zusätzlichen Abstand zwischen zwei Zeilen innerhalb eines Absatzes oder einer Tabelle erreicht man mit dem Befehl `\[länge]`:

`\[1ex]`, `\[15mm]`

7. Schriftformatierung

Kurzbeschreibung
S. 41–44
Not So Short Intro
p. 107-111

7.1 Schriftgröße

Schriftgrößen und Zeilenabstand (`\baselineskip`) in den Standard-Dokumentklassen:

Schriftgrößenbefehl	Grundschriftgröße					
	[10pt]		[11pt]		[12pt]	
<code>\tiny</code>	5	6	6	7	6	7
<code>\scriptsize</code>	7	8	8	9.5	8	9.5
<code>\footnotesize</code>	8	9.5	9	11	10	12
<code>\small</code>	9	11	10	12	10.95	13.6
<code>\normalsize</code>	10	12	10.95	13.6	12	14.5
<code>\large</code>	12	14	12	14	14.4	18
<code>\Large</code>	14.4	18	14.4	18	17.28	22
<code>\LARGE</code>	17.28	22	17.28	22	20.74	25
<code>\huge</code>	20.74	25	20.74	25	24.88	30
<code>\Huge</code>	24.88	30	24.88	30	24.88	30

Diese Befehle sind Schalter, d. h. sie wirken auf den gesamten folgenden Text. Um die Wirkung solcher Schalter auf einen Textabschnitt zu begrenzen, verwendet man `{ }`, z.B. so:

```
{\small <some text>}
```

Der Abstand im gesamten Absatz richtet sich nach der größten Schrift. Wenn kein Absatzende vorkommt, wird jedoch der Zeilenabstand nicht angepasst. Deshalb im Bedarfsfall z. B. schreiben: `{\Huge ... \par}`

7.2 Weitere Schriftmerkmale

`\fontencoding` (wird selten benötigt, i. d. R. **T1**)
`\fontfamily` **rm** Roman **sf** Sans-Serif, **tt** Teletype
`\fontseries` **md** medium (**m**), **bf** boldface [**fett**] (**b** od. **bx**)
`\fontshape` **upright** (**n**), **italic** [*kursiv*], **slanted** [*geneigt*], **sc** SMALL CAPS
 [KAPITÄLCHEN]

Die **Schriftfamilie** sollte i. a. innerhalb eines Dokuments nicht gewechselt werden. Die Standardschrift ist über den Befehl `\familydefault` festgelegt, wobei dieser mit `\rmdefault` vordefiniert ist. Um auf eine Sans-Serif als Standardschrift umzustellen, muss man (i. d. R. in der Präambel) den Befehl

```
\renewcommand{\familydefault}{\sfdefault}
```

 eingeben.

WICHTIG: Serie und Form können

- ▷ entweder durch *Schalter* wie `\bfseries`, `\itshape` verändert werden (ggf. innerhalb von `{ }`),
- ▷ oder man kann **Befehle** verwenden wie `\textbf{text}` `\textit{text}` etc.

Achtung, nicht alle Schriften können alles. Es gibt z. B. Schriften, die keine *fett-kursiven* Glyphen beinhalten.

7.3 Schriftarten (Fonts)

Standardmäßig wird in \LaTeX eine von Donald Knuth entworfene Schrift namens *Computer Modern* (CM) verwendet. Es gibt aber inzwischen Hunderte oder Tausende von Computerschriften. Viele davon sind kommerziell, aber viele auch frei.

Schriften sind Software. Zur Definition von Schriften gibt es verschiedene Methoden:

- Bitmap-Fonts. In (La) \TeX mit Hilfe von METAFONT in der gewünschten Größe erzeugt. Veraltet.
- Vektorfonts (CM ist im Hintergrund als Vektorfont definiert).
- Outline-Fonts
 - *Postscript-Type 1*. Standard auf Unix-/Linuxsystemen und in \LaTeX . Die CM-Schrift ist auf modernen \LaTeX -Systemen als Type-1-Schrift implementiert. Eine Schrift besteht aus 2 bis 6 Dateien:
 1. `.pfm` (PostScript Font Metric),
 2. `.pfb` (PostScript Font Binary, auch als `.pfa` in ASCII statt Binary),
 3. `.inf` (optional, wird nur für Windows benötigt),
 4. `.afm` (Adobe Font Metrics, optional), und evt. noch
 5. `.fd` (font definition, Zuordnung der Schriftschnitte in \TeX) und
 6. `.map` (Äquivalenz interne und \TeX -Schriftnamen).
 - *TrueType*. Ursprünglich von Apple entwickelt, um nicht von Adobe abzuhängen, dann auch von Microsoft übernommen. Nur eine Datei (`.ttf`).
 - *OpenType*. Weiterentwicklung von Microsoft und Adobe, der aktuellste Standard. Können in \LaTeX nutzbar gemacht werden, in $X\TeX$ und $\text{Lua}\TeX$ standardmäßig verfügbar.

7.3.1 Zur Schriftauswahl

Agmi Agmi

- Serifenschrift:
 1. Zeile ist optisch nach unten abgeschlossen, erleichtert den Lesefluss vor allem bei längeren Zeilen
 2. Wird daher fast überall als "Brottschrift" für längere Texte verwendet.
 3. Die Serifen werden bei kleinen Schriften / Ausgabegeräten mit schlechter Auflösung nicht mehr einwandfrei dargestellt. Auch Scannen oder Kopieren tut ihnen nicht so gut.
- Serifenlose Schrift
 1. Aus o. g. Gründen bei klein gedruckten Texten vorzuziehen.
 2. Für Lesen am Bildschirm, aber auch für Folder, Plakate, evt. Überschriften
 3. Wegen des unruhigeren Schriftbilds mehr Zeilenabstand erforderlich
 4. Wirkt moderner, Nationalschrift der Schweiz („Helvetica“)

- ▶ Achtung, viele Schriften haben keine Glyphen für Mathematik-Satz! L^AT_EX ersetzt in diesem Fall fehlende Zeichen aus der CM-Schrift. Besonders bei serifenlosen Schriften ist das freie Angebot von math-Schriften nicht groß.
- ▶ Achtung, im PDF muss die vollständige Schriftinformation eingebettet sein, sonst kann das File auf Rechnern, auf denen die jeweilige Schrift fehlt, weder (korrekt) angezeigt noch gedruckt werden! Vorgeschrieben für Archivstandard PDF-1A, z. B. für digitale Ablieferung von Abschlussarbeiten.
- ▶ Ehemals „Postscript-Standardschriften“ (mit Ghostscript und vielen Druckern mitgeliefert):
 - **Helvetica**. Die einzige Sans-Serif-Schrift. Der Arial-Schrift von Microsoft sehr ähnlich. Relativ groß, bei Mischverwendung auf 95% skalieren. Name: phv. Paket: helvet
 - **Courier**. Eine klassische Schreibmaschinenschrift. Solche Schriften braucht man, wenn man Code darstellen will. Jedes Zeichen ist gleich breit, dadurch passen zum Beispiel Einrückungen. Auch sollten z.B. l, I und 1 gut unterscheidbar sein, sowie 0 und O. Dünner als die anderen Schriften, nicht ideal in Kombination. Name pcr. Paket: courier, couriers
 - **Times-Roman**. Extrem weit verbreitet. Eigentlich für Zeitungen, läuft eng, sehr feine Serifen, daher als Standardschrift nicht ideal. Math: ja (gut). Mit eigener tt-Schrift. Name: ptm. Paket: txfonts u. a.
 - **Palatino**. Vom berühmten Schriftdesigner Hermann Zapf entworfen, gerne für Bücher verwendet, Serifen nicht zu fein. Math: ja. Mit eigener tt-Schrift. Name: ppl. Paket: pxfonts u. a.
 - **Bookman**. In Amerika verbreitet. Boldface sehr fett. Läuft weit, gut lesbar. Math: nein. Name: pbk. Paket: tgbonum
 - **NewCentury-Schoolbook**. Eine echte Schulbuch-Schrift, sehr gut lesbar, aber größere Laufweite. Math: ja (Fourier). Name: pnc. Paket: fouriernc
- ▶ Einige weitere Alternativen:
 - **Arev**. Hat Math-Support, zum Beispiel für Folien oder Poster. Passende tt-Schrift inkludiert. Name: fav, fvm, ... Paket: arev
 - **Computer Concrete**. Schriftform mit Serifen, aber alle gleich dick. Ein möglicher Kompromiss Bildschirm – Drucker, mit Math-Support. Name: ccr, ccm, ... Paket: concmath.

The L^AT_EX Font Catalogue: <http://www.tug.dk/FontCatalogue/>

Liste der Kurznamen vieler Schriften: <https://www.tug.org/fontname/html/Font-name-lists.html> (jeweils nur die ersten drei Buchstaben verwenden!).

7.3.2 Verwendung von Schriftarten

In der Regel über das entsprechende Paket und dann ggf. Befehle für die Schriftfamilie.

Zum Beispiel: `\usepackage{txfonts}` um die Times-Roman (samt assoziierten sf- und tt-Schriften) zu laden.

Muss man *ausnahmsweise* die Schriftart selbst innerhalb eines Dokuments wechseln, so geht das mit

```
\fontencoding{..}
\fontfamily{..}\fontseries{..}\fontshape{..}
\fontsize{..}{..}\linespread{..}
\selectfont
```

Als Argument von `\fontfamily` ist der o.g. Name, z.B. `phv` anzugeben. Die `\fontsize{}{}` Angabe kann dazu verwendet werden, **beliebige Schriftgrößen** zu nutzen. Das zweite Argument ist dann `\baselineskip`.

Argumente der anderen Befehle (selten erforderlich) sind 1 oder 2 Buchstaben, siehe oben (wo eine **blaue** Version angegeben ist, ist diese zu verwenden).

Für alle Details zur Auswahl von Schriften und Schrifteigenschaften siehe <http://www.latex-project.org/guides/fntguide.pdf>.

7.4 Typografische Feinheiten

Italic-Korrektur: am Ende einer Sequenz mit Umschaltung auf `\itshape`

[*Das Schiff*] [*Das Schiff*]

```
{\Large [{\itshape Das Schiff}] [{\itshape Das Schiff\}]
```

Nicht erforderlich bei Verwendung des Befehls `\textit{text}`

Ligaturen **fi** aufheben **fi**:

- ▷ `\`
- ▷ `\kern0pt`
- ▷ `"|` bei Verwendung einer deutschen Sprachoption, setzt zugleich Trennstelle für Silbentrennung
- ▷ `\usepackage{microtype} \DisableLigatures[f]{encoding = T1}`
Die Angabe von `[f]` bewirkt, dass nur Ligaturen mit "f" betroffen sind, ohne `[f]` sämtliche.

7.5 Spezielle Schriftauszeichnungen

7.5.1 Unterstreichen

`\underline{text}`: Text mit Unterlänge

7.5.2 Durchstreichen, Unterwellen, etc.

Paket **ulem**:

- ▷ `\uline{important}` important text
- ▷ `\uuline{urgent}` urgent
- ▷ `\uwave{boat}` boat
- ▷ `\sout{wrong}` ~~wrong~~

- ▷ `\xout{removed}` ~~`\xout{removed}`~~
- ▷ `\dashuline{dashing}` `dashing`
- ▷ `\dotuline{dotty}` `dotty`

7.5.3 Text sperren

Benötigt das Paket `microtype`

`\textls[amount]{Beispielstext}` vs. Beispielstext

(1s – englisch ‘letterspacing’)

amount ist die Sperrweite in 1/1000em. Default ist 100 (d. h. 0.1em)

7.6 Zusammenfassung: Wie gehe ich vor, um die Schriftart und -formatierung zu ändern?

Rule 1: Load a font package if you want to change the font.

Hint: Some font packages change all three families (rm, sf, tt), others only some.

For example, the `helvet` package only changes the sf-family, while the `pxfonts` package changes all three.

Rule 2: If you want to change the base font family of your document from `rm` to `sf`, add the following line to your header:

```
\renewcommand{\familydefault}{\sfdefault}
```

Hint: Some font packages will already do this for you. Then you don’t need that.

Hint: This applies even for the default font (that you have if you don’t load a font package), *Computer Modern*.

Hint: `\familydefault` is the default family of your document. It can be one of

```
\rmdefault \sfdefault \ttdefault
```

Hint: These three internal commands can be reassigned. For example, the `pxfonts` package contains:

```
\renewcommand{\rmdefault}{pxr}
\renewcommand{\sfdefault}{pxss}
\renewcommand{\ttdefault}{pxtt}
```

A user will rarely need to do this.

Rule 3: Do not use low-level font commands unless you really need to and you know what you do!

Hint: Low-level font commands are

(i) those three listed above, and

(ii) any command that requires `\selectfont` to become effective (see Handout 3, 3.3.2).

8. Boxen

8.1 Das Box-Konzept in \LaTeX

\TeX verpackt jegliches Material in Boxen und ordnet diese in Zeilen an; eine Zeile gibt wiederum eine Box, und die Zeilenboxen werden in einen Absatz angeordnet, der auch wieder eine Box ist . . .

Daher gibt es Kommandos für Boxen; diese können für bestimmte Aufgaben sehr nützlich sein.

Not So Short Intro
p. 118–120

8.2 Links-Rechts-Box (LR-Box)

In dieser Art von Box kann kein Zeilenumbruch stattfinden. Das ist auch ein Grund sie zu verwenden, wenn man möchte, dass ein Wort oder eine Wortgruppe nicht getrennt wird.

Wenn keine Breite angegeben, ist die Breite die natürliche.

- ▶ `\mbox{text}` simpelste Form
- ▶ `\fbox{text}` wie `\mbox{}`, aber eingerahmt.
 - Rahmendicke: `\setlength{\fboxrule}{length}` eingerahmt mit 2pt-rule
 - Abstand: `\setlength{\fboxsep}{length}` eingerahmt mit 0.3pt-sep
- ▶ `\makebox[width][position]{text}` ohne optionale Parameter wie `\mbox{}`.
 - width: Breite der Box. `\width` ist die natürliche Weite entsprechend `text`.
 - position: `center`, `flushleft`, `flushright`, or `spread the text to fill the box`.
- ▶ `\framebox[width][position]{text}` analog
- ▶ `\usepackage{fancybox}` ermöglicht eingerahmte “fancy boxes”, z. B. mit Schatten, oder um besondere Inhalte.
- ▶ `\rule[lift]{width}{thickness}` füllt Box mit Druckerschwärze: `\rule[.5ex]{2cm}{2pt}` ergibt ,
 oder `\rule{10pt}{1cm}`  Oft ist es nützlich, `rule` mit Länge oder Breite 0pt zu verwenden.
- ▶ `\strut` „Stütze“, um eine Box mit der Höhe wie ein Text mit Ober- und Unterlängen + Rand zu erzeugen: `\rule{1em}{.2pt}\fbox{\strut}\fbox{A}\fbox{g}\rule{1em}{.2pt}`
- ▶ `\raisebox{lift}[extend-above-baseline][extend-below-baseline]{text}` Box um `lift` vertikal nach oben oder unten (negativer Wert) versetzen. Mittels der `extend`-Parameter kann die natürliche Größe der Box durch die entsprechenden Werte ersetzt werden.
 Folgende interne Parameter können (hier, aber auch sonst) nützlich sein:
`\width`, `\height`, `\depth`, `\totalheight`

Verwandt:

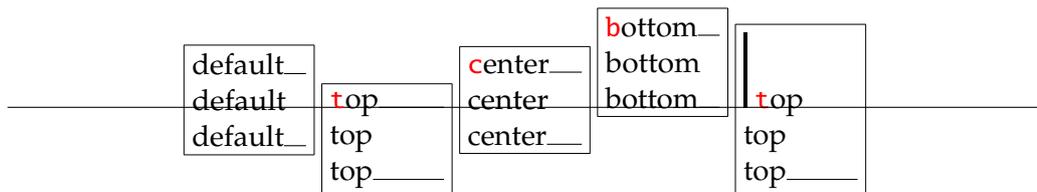
- `text`
Schreibt `text` hochgestellt und kleiner (!), nützlich für Einheiten mit Exponent, wenn man nicht `SIunitx` (mehr dazu später) verwendet, oder 2^{nd} , M^{me} und dergleichen.
- `\textsubscript{text}` Analog, braucht Paket `fixltx2e`. Selten erforderlich, für chemische Summenformeln besser eigenes Paket (siehe später).
- ▶ `\hrulefill` füllt den Rest der Zeile mit einer Linie: _____
- ▶ `\dotfill` detto aber mit einer Punktlinie:

8.3 Absatzboxen (parbox)

8.3.1 Parbox

`\parbox[position][height][inner-pos]{width}{text}`

- ▶ Eine Breite muss immer angegeben werden, Inhalt wird dann auf diese Breite umgebrochen.
- ▶ `position` kann `center`, `top`, `bottom` sein. Dies bezeichnet die Stelle, mit der die Box in den umgebenden Text eingehängt wird. Beispiel:



- ▶ `height` fixe Höhe (normalerweise passt sich die Höhe automatisch dem Inhalt an). Kann nur verwendet werden, wenn `position` angegeben.
- ▶ `inner-pos` nur wenn fixe Höhe gegeben, für vertikale Anordnung des Inhalts. Mögliche Werte: `c t b s`

8.3.2 Minipage

`\begin{minipage}[position][height][inner-pos]{width}`
`text`
`\end{minipage}`

Die Minipage funktioniert fast wie eine Parbox. Gedanklich kann eine Parbox als ein oder mehrere Absätze gesehen werden, eine Minipage als eine kleine Seite (wobei die Minipage Leerraum am Anfang entfernt, die Parbox nicht.) `verbatim`-Text kann nur in der Minipage verwendet werden.

8.3.3 Verwendung von parbox und minipage

- ▶ Anordnung von mehrzeiligen Inhalten nebeneinander.
- ▶ Rahmen um mehrzeilige Inhalte zeichnen:
`\fbox{\parbox{width}{text}}`

- ▶ Seitenumbrüche sind weder in einer Minipage noch in einer Parbox möglich. Wenn eingerahmter Text über mehrere Seiten gebraucht wird: Pakete `framed` oder `mdframed`.
- ▶ Fußnoten sind in der Parbox nicht möglich, und erscheinen bei der Minipage am Fuß derselben (mit Paket `ftn` regulär am Fuß der Hauptseite).
- ▶ Zur Angabe der Breite sind einige interne Längen von \LaTeX hilfreich:
 - `\textwidth` Breite des Textblocks
 - `\paperwidth` Breite der Seite inkl. Ränder
 - `\columnwidth` Breite der Textspalte (wenn mehrspaltig)
 - `\linewidth` Breite der Zeile in der aktuellen Umgebung
 Diese Werte können mit Faktoren versehen werden, z. B. 0.48\textwidth für eine Box, die knapp schmaler ist als der halbe Textblock, so dass zwei solche Boxen mit etwas Zwischenraum nebeneinander passen.

8.3.4 Ausgabe und Manipulation von Längen

Die Werte von Längen (und Zählern) können mit dem \TeX -Befehl `\the` ausgegeben werden, z. B. `\the\linewidth`.

Für selbstdefinierte Längen und die Veränderung von solchen Längenvariablen mittels \TeX siehe

https://en.wikibooks.org/wiki/LaTeX/Lengths#Length_manipulation

8.4 Besondere Boxen

Nur mit `\usepackage[driver]{graphics}`. Dokumentation:

<http://mirrors.ctan.org/macros/latex/required/graphics/grfguide.pdf>

Alle diese Boxen sind LR-Boxen (wie `mbox`)!

- ▶ `\rotatebox[key-val-list]{angle}{text}`
Rotiert um linkes unteres Eck der Box. Für anderen Ursprung siehe Dokumentation (key-val-list, p. 7, z. B. `origin=c`).
- ▶ `\scalebox{hscale}[vscale]{text}`
 $-\infty \leq [hv]scale \leq \infty$. Wenn kein `vscale`: `vscale=hscale`
- ▶ `\reflectbox{text}`
An abbreviation for `\scalebox{-1}[1]{text}`
- ▶ `\resizebox{hlength}{vlength}{text}`
Skalierung auf fixe Größe. Wenn nur eine Länge vorgegeben und die andere Dimension den gleichen Skalierungsfaktor verwenden soll, für die andere Länge ! angeben.
- ▶ `\colorbox` wird später behandelt (Verwendung von Farben).

9. Texte in anderen Sprachen als Englisch

9.1 Was ist sprachspezifisch in L^AT_EX?

- Silbentrennmuster (hyphenation patterns)
- Textbausteine wie *Chapter*, *Figure*, das Datum etc.
- Spezielle Glyphen (Umlaute etc.) können leichter zugänglich gemacht werden.

9.2 Sprachen wählen

Die Standardsprache in L^AT_EX ist (US-)Englisch. Für die Verwendung anderer Sprachen wird das Paket `babel` gebraucht:

```
\usepackage[naustrian,UKenglish]{babel}
```

- In der Argumentliste kann eine oder mehrere Sprachen angegeben werden.
- Für die deutsche Sprache gibt es vier Optionen: `german`, `ngerman`, `austrian`, `naustrian`. Die `n`-Version verwendet die Trennregeln der neuen deutschen Rechtschreibung (seit 2005 verbindlich). `austrian` unterscheidet sich von `german` dadurch, dass bei Ausgabe des Datums (mit `\today` oder implizit in `\maketitle`) statt Januar *Jänner* steht.
- Die *letzte* angegebene Sprache wird automatisch eingeschaltet.
- Sprache wechseln:
 - Umschalten der Sprache mit `\selectlanguage{english}`
 - Umgebung für begrenzten Text in anderer (hier: englischer) Sprache:


```
\begin{otherlanguage}{english} ...
\end{otherlanguage}
```
 - Kurzer Text in anderer Sprache `\foreignlanguage{english}{...}`
- `english` ist amerikanisches English, es ist aber auch `USenglish` (synonym) oder `british` (`UKenglish` als Synonym) möglich. Bewirkt andere Schreibung des Datums und etwas andere Trennregeln.

9.3 Eingabe von (deutschen) Sonderzeichen

Kurzbeschreibung
S. 21–22

Zeichen	Standard- version	Shorthands mit (n)german	Direkteingabe mit inputenc
ä	<code>\"a, \"{a}</code>	<code>"a</code>	ä
ß	<code>{\ss}, \ss{}</code>	<code>"s</code>	ß
„...“	<code>\glqq \ldots \grqq{}</code> ⁺	<code>"' \ldots "'</code>	– [*]
„...‘	<code>\glq \ldots \grq\</code> ⁺	–	– [*]
”	<code>\textquotedbl#</code>	<code>\dq</code>	”
»...«		<code>">\ldots"<</code>	
«...»	<code>\guillemotleft~/right</code>	<code>\og~ldots~\fg</code> [§]	

⁺ nur mit deutscher Sprache aktiv!

^{*} nicht auf Keyboard! Evt. autom. Ersetzung von ” durch Editor

[#] mit Paket `textcomp`

[§] französische Anführungszeichen, mit `[french]{babel}`

Für mehr Details zu Französisch und weiteren Sprachen siehe z. B.

<https://en.wikibooks.org/wiki/LaTeX/Internationalization#French>

10. Absatzformatierung

L^AT_EX setzt standardmäßig (außer in letter) alle Absätze wie folgt:

- ▷ Blocksatz (justified)
- ▷ Erste Zeile eines jeden Absatzes um `\parindent` eingezogen.
 - ausgenommen erster Absatz nach Überschrift
 - Bei 11pt Grundschrift: `\parindent=0.0pt=\the\parindent`
- ▷ Kein Abstand zwischen Absätzen (d.h. `\parskip 0pt`)
- ▷ Innerhalb einer parbox oder minipage wird `\parindent` auf 0pt gesetzt!

10.1 Einzug und Absatzabstand ändern

`\noindent` verhindert Einzug für den Absatz, an dessen Beginn es steht. (`\indent` erzeugt Einzug.)

`\parindent0pt` setzt den Einzug generell auf Null. Dann sollte man aber einen Absatzabstand definieren, sonst sind Absätze nicht mehr erkennbar:

Zum Beispiel `\parskiplex` oder `\parskip 1ex plus 0.5ex`, oder `\parskip\medskipamount`. Oder Paket `parskip` verwenden (setzt `\parskip=0.5\baselineskip \advance\parskip by 0pt plus 2pt`).

10.2 Ausrichtung ändern

Kurzbeschreibung
S. 22

Ausrichtung	Schalter	Umgebung	mit <code>ragged2e</code>	
Blocksatz (justified)	<code>\raggedcenter</code> ¹	–	<code>\justifying</code>	<code>{justify}</code>
Flattersatz linksbündig	<code>\raggedright</code>	<code>{flushleft}</code> ²	<code>\RaggedRight</code>	<code>{FlushLeft}</code>
Flattersatz rechtsbündig	<code>\raggedleft</code>	<code>{flushright}</code> ²	<code>\RaggedLeft</code>	<code>{FlushRight}</code>
Zentriert	<code>\centering</code>	<code>{centered}</code>	<code>\Centering</code>	<code>{Center}</code>

¹In manchen Dokumentationen angegeben, aber offensichtlich nicht implementiert.

²`raggedright` und `raggedleft` als Umgebungsnamen scheinen auch zu funktionieren.

L^AT_EX schaltet bei allen Ausrichtungen außer Blocksatz die Silbentrennung ab. Das ist z. B. für Überschriften sinnvoll, sonst aber oft störend. Abhilfe durch das `ragged2e`-Paket. Es hat die optionalen Parameter

- ▷ `[raggedrightboxes]` All parboxes, minipages, marginpars and p-columns of tabulars and arrays are automatically set using `\RaggedRight`.
- ▷ `[footnotes]` This options sets all footnotes ragged-right by loading the `footmisc` package with the `ragged` option.

10.2.1 Umgebungen für Zitate und Verse

Kurzbeschreibung
S. 20

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

Absätze von rechts und links eingerückt. Die `quotation`-Form verwendet die originalen Standardwerte von `\parindent` und `\parskip` (auch wenn diese im Header umdefiniert sind!), die `quote`-Form jene, die aktuell definiert sind.

`\begin{verse} ... \end{verse}`:

Für Gedichte und dergleichen.

Ähnlich `quote`, aber ohne `\parindent`.

Jede Zeile ist mit `\\` zu beenden, wenn eine Gedichtzeile länger ist als eine Textzeile wird sie mit hängendem Einzug umgebrochen.

10.3 Umgebungen für Code und dergleichen

Kurzbeschreibung
S. 22

Alles wörtlich, Steuerzeichen ignorieren, in `tt`-Schrift: `\verb|text|`
für mehrzeiliges Material: `\begin{verbatim} ... \end{verbatim}`.

Achtung, auch Zeilenumbrüche werden exakt übernommen
(mit Paket `spverbatim` wird Umbruch erlaubt)!

Siehe auch Pakete `verbatim`, `newverbs`, `shortvrb`, `moreverb`, `fancyvrb`.
`verbatim` hat auch ein `\begin{comment}content... \end{comment}` zum Auskommentieren von längeren Abschnitten (vor allem wenn man kein `TeXstudio` o.ä. hat).

Zum Ausgeben von Source code einer Programmiersprache gibt es auch das `listings`-Paket ("pretty-print"-Funktionalität) sowie diverse andere Pakete (siehe <https://www.ctan.org/topic/listing>). `listings` ist vielseitig anpassbar, besonders empfehlenswert ist `basicstyle=\ttfamily` zu verwenden.

10.4 Zeilenabstand ändern

Kurzbeschreibung
S. 39

Begriffe:

- Zeilendurchschuss (leading)
- Zeilenabstand (line spacing, `\baselineskip`)

Quelle Abb.: Wolfgang Beinert,
<http://www.typolexikon.de/>



`\baselineskip` sollte nicht explizit verändert werden, statt dessen mit `\linespread{factor}` (in der Präambel, wirkt global) anpassen.

Noch besser: Paket `setspace` verwenden:

- Globale Optionen (Präambel): `\singlespacing`,
`\onehalfspacing`, `\doublespacing`, `\setstretch{baselinestretch}`
- Umgebungen mit den Namen
`{singlespace}`, `{onehalfspace}`, `{doublespace}`
- Umgebung mit beliebigem Zeilenabstand:
`\begin{spacing}{baselinestretch} .. \end{spacing}`

`baselinestretch` ist ein beliebiger Faktor.

10.5 Silbentrennung

L^AT_EX ist recht gut mit der (automatisch eingeschalteten) Silbentrennung, es ist nur selten erforderlich, manuell nachzuhelfen. Es gibt im Wesentlichen zwei Fälle, wo das erforderlich sein kann:

- ▶ Wörter mit Bindestrich (werden außer am Bindestrich nicht getrennt)
- ▶ ausgefallene Wörter, Fachbegriffe, Wörter in Fremdsprachen (die nicht mit `\otherlanguage` o. ä. der richtigen Sprache zugeordnet wurden)

10.5.1 Trennstellen vorgeben

- ▶ An Ort und Stelle mit `\-`. Da bei Angabe einer solche Stelle an anderen Stellen nicht mehr von selbst getrennt wird, sollte man immer alle Trennstellen angeben, z. B. `Trenn\ -stel\ -le`.
- ▶ Wenn heikle Wörter öfter vorkommen, in der Präambel mittels einer Wortliste `\hyphenation{Trenn-stel-len ...}`
Achtung, Wörter in der Wortliste dürfen keine Akzente enthalten, mit `inputenc` arbeiten!

10.5.2 Weitere Hilfen

- ▶ Trennung verhindern: in `\mbox{...}` einschachteln.
- ▶ `\sloppy` (schlampig): Schalter, der weiteres Auseinanderziehen von Wörtern in einer Zeile erlaubt.
Nützlich für enge Spalten, in denen trotz Silbentrennung viele übervolle Zeilen auftreten. `\fussy` (pingelig) schaltet wieder in den Standardmodus zurück. Auch als Umgebung verwendbar.
Mit `\setlength{\emergencystretch}{1em}` kann man je nach Wert sanfter eingreifen (ggf. nach heikler Textstelle wieder auf 0em setzen).
- ▶ Für Trennung in URLs das `url`-Paket und `\url{...}` verwenden. Es trennt standardmäßig nicht an Bindestrichen, das kann aber mit der Option `[hyphens]` erlaubt werden. Siehe auch Paketdokumentation.
- ▶ `"~` vor Trennung geschützter Bindestrich
- ▶ `"-` Bindestrich, nach dem getrennt werden darf, und der auch noch weitere Standardtrennstellen ermöglicht
- ▶ `""` ermöglicht Trennung an dieser Stelle, ohne einen Trennstrich einzufügen.
- ▶ `\slash` erzeugt einen /, nach dem ein Zeilenumbruch erfolgen kann
- ▶ `\renewcommand{\lefthyphenmin}{2}` verhindert eine Trennung, bei der ein einzelner Buchstabe auf der vorherigen Zeile bleibt. Kann auch nützlich sein, wenn man im Flattersatz mit dem `ragged2e`-Paket Trennungen nur dann zulassen will, wenn sie „wirklich gebraucht“ werden (hier entsprechend großen Zahlenwert nehmen).

Achtung, die Methoden mit `"` gelten nur im deutschen Sprachmodus mit dem `babel`-Paket! Man kann die deutschen Shorthands auch in englischen Texten verfügbar machen, siehe <http://tex.stackexchange.com/questions/27198/babel-adding-ngerman-s-language-shorthands-to-english-as-the-main-document-lan> (kompliziert, aber man kann dadurch mehr über T_EX lernen.)

11. Neu- und Umdefinition von Befehlen und Umgebungen

Befehle und Umgebungen können in der Präambel mit globaler Gültigkeit oder später neu definiert werden mit

`\newcommand{cmd}[nargs]{def}` wobei

- `cmd` der Name des neuen Befehls ist; muss mit `\` beginnen und darf nur Buchstaben enthalten (Groß- / Kleinschreibung signifikant, keine Zahlen und Sonderzeichen!)
- `nargs` ist die Anzahl der Argumente ($1 \leq \text{nargs} \leq 9$, nicht angeben wenn keine Argumente)
- `def` ist die Definition für den Befehl; Platzhalter für die Argumente sind dabei `#1, #2, ...`

Beispiel für Text fett und kursiv:

`\newcommand{\textbi}[1]{\{\bfseries\itshape #1}}`

Anwendung: `\textbi{text}` ergibt ***Text***

Ein bereits bestehender Befehl (egal ob system- oder benutzerdefiniert) kann mit dem analog aufgebautem `\renewcommand` umdefiniert werden.

Analog können auch Umgebungen neu definiert oder umdefiniert werden:

`\renewenvironment{name}[args]{begdef}{enddef}`

wobei `begdef` für den Teil steht, der vor dem Inhalt der Umgebung kommt, und `enddef` für den nachher. NB: Die Argumente könne nur im `begdef`-Teil verwendet werden!

Zum Beispiel könnte man (wenn `paralist` geladen ist) die kompakte Aufzählung kürzer zugänglich machen mit

`\newenvironment{ci}{\begin{compactitem}}{\end{compactitem}}`

(ggf. zur Übersichtlichkeit auf mehrere Zeilen aufteilen, die mit `%` abgeschlossen sind.)

12. Aufteilung eines Dokuments auf mehrere Dateien

Not So Short Intro
p. 14

Bei umfangreichen Dokumenten, wie etwa einem Buch oder einer größeren Abschlussarbeit, kann es sinnvoll sein, den Quellcode auf mehrere *.tex-Files zu verteilen, zum Beispiel kapitelweise.

Auch bei Abbildungen und Tabellen, deren Inhalt von externen Programmen erzeugt wird (Details dazu in Kap. 14 und 15), ist es manchmal nützlich, diesen aus einem File einzubinden, so dass bei einer Änderung im Textfile nichts aktualisiert werden muss und es reicht, das Dokument neu zu übersetzen.

Zu diesem Zweck gibt es zwei Befehle: `\input{file}` und `\include{file}`.

12.1 `\input`

Dies ist der einfachere der beiden Befehle, mit der Syntax `\input{file}`.

Falls der Filename auf .tex endet, braucht die Endung nicht angegeben werden, andernfalls ist auch die Endung frei wählbar. Der Inhalt des Files wird an der jeweiligen Stellen 1:1 eingefügt. Man kann dies auch dazu nützen, die Präambel auszulagern. In einer solchen Datei können weitere `\include{file}`-Befehle enthalten sein.

12.2 `\include`

Die Syntax ist analog `\include{file}`. Der Befehl beginnt automatisch eine neue Seite, ist also in der Regel nur für ganze Kapitel in einem report oder book sinnvoll, bei denen man für jedes Kapitel eine eigene Datei erstellt.

Der Vorteil gegenüber `\input` ist, dass man – nachdem man einmal das Gesamtdokument erzeugt hat, so dass alle aux-, toc-, etc. Dateien mit den Verweisen erstellt sind – in die Präambel den Befehl

```
\includeonly{file1, file2, ..., file-n}
```

aufnehmen kann. Aus der Liste kann man beliebig Files entfernen; dann werden beim nächsten Übersetzen nur mehr die genannten Kapitel genommen, die Verweise auf andere Kapitel oder deren Inhalt funktionieren aber weiter und verursachen keine Fehlermeldungen. Man kann z. B. nur das Kapitel inkludieren, an dem man aktiv arbeitet, und muss beim testweisen Übersetzen nicht warten, bis wieder das ganze Dokument aktualisiert ist.

Zu beachten ist, dass man den Befehl nicht für Material in der Präambel verwenden kann, und nicht verschachtelt werden darf (inkludierte Dateien können aber beliebig von `\input` Gebrauch machen).

13. Listen

Kurzbeschreibung
S. 21

13.1 Eingebaute Listen

L^AT_EX kennt drei (mit `trivlist` vier) Listenumgebungen:

- **itemize**
 - verschachtelt
 - * noch mehr
 - innerste

1. **enumerate**
2. usw.
 - (a) verschachtelt
 - i. noch mehr
 - A. innerste

description Erklärung Erklärung Erklärung Erklärung Erklärung Erklärung Erklärung Erklärung Erklärung
Echter Eintrag Erklärung

Die Syntax ist immer

```
\begin{listentyp}
  \item[individuelle Marke] Text
\end{listentyp}
```

`\item` wird in T_EXstudio mit `<shift><ctrl><i>` als Shortcut erzeugt.

Es gibt viele Möglichkeiten, das Layout dieser Listen zu verändern.

Die Symbole ("label") der `itemize`-Umgebung sind standardmäßig wie folgt definiert:

Ebene	Befehl cmd	Voreinstellung def	Symbol
1	<code>\labelitemi</code>	<code>\textbullet</code>	•
2	<code>\labelitemii</code>	<code>\normalfont\bfseries\textendash</code>	–
3	<code>\labelitemiii</code>	<code>\textasteriskcentered</code>	*
4	<code>\labelitemiv</code>	<code>\textperiodcentered</code>	·

Die Labels können mit `\renewcommand{cmd}{def}` geändert werden.

Interessant ist, dass man die Zähler von `enumerate` mit `\setcounter{counter}{value}` setzen kann, z. B. um eine vorher begonnene Liste mit an anderer Stelle fortzusetzen. Dazu kann man auch den letzten Wert speichern mit

Die Nummerierung in der `enumerate`-Umgebung funktioniert wie folgt:

Ebene	Zähler counter	Label	Voreinstellung def	Bsp.
1	<code>enumi</code>	<code>\labelenumi</code>	<code>\arabic{enumi}</code> .	1.
2	<code>enumii</code>	<code>\labelenumii</code>	<code>(\alph{enumii})</code>	(a)
3	<code>enumiii</code>	<code>\labelenumiii</code>	<code>\roman{enumiii}</code> .	i.
4	<code>enumiv</code>	<code>\labelenumiv</code>	<code>\Alph{enumiv}</code> .	A.

`\newcounter{enumilast}\setcounter{enumilast}{\value{enumi}}` und mit `\setcounter{enumi}{\value{enumilast}}` am Beginn der neuen Liste wieder zuweisen kann. Der Wert (value) eines Zählers counter kann mit `\thecounter` in Ziffern ausgegeben werden, aber auch mit `\arabic{counter}`, `\Roman{counter}`, etc.

description-Liste

Man kann die Labels individuell formatieren oder mit Zusätzen versehen durch `\renewcommand{\descriptionlabel}[1]{\hspace{\labelsep}...#1...}` wobei die ... für die individuellen Formatierungsanweisungen etc. stehen.

13.2 Pakete zur Listengestaltung

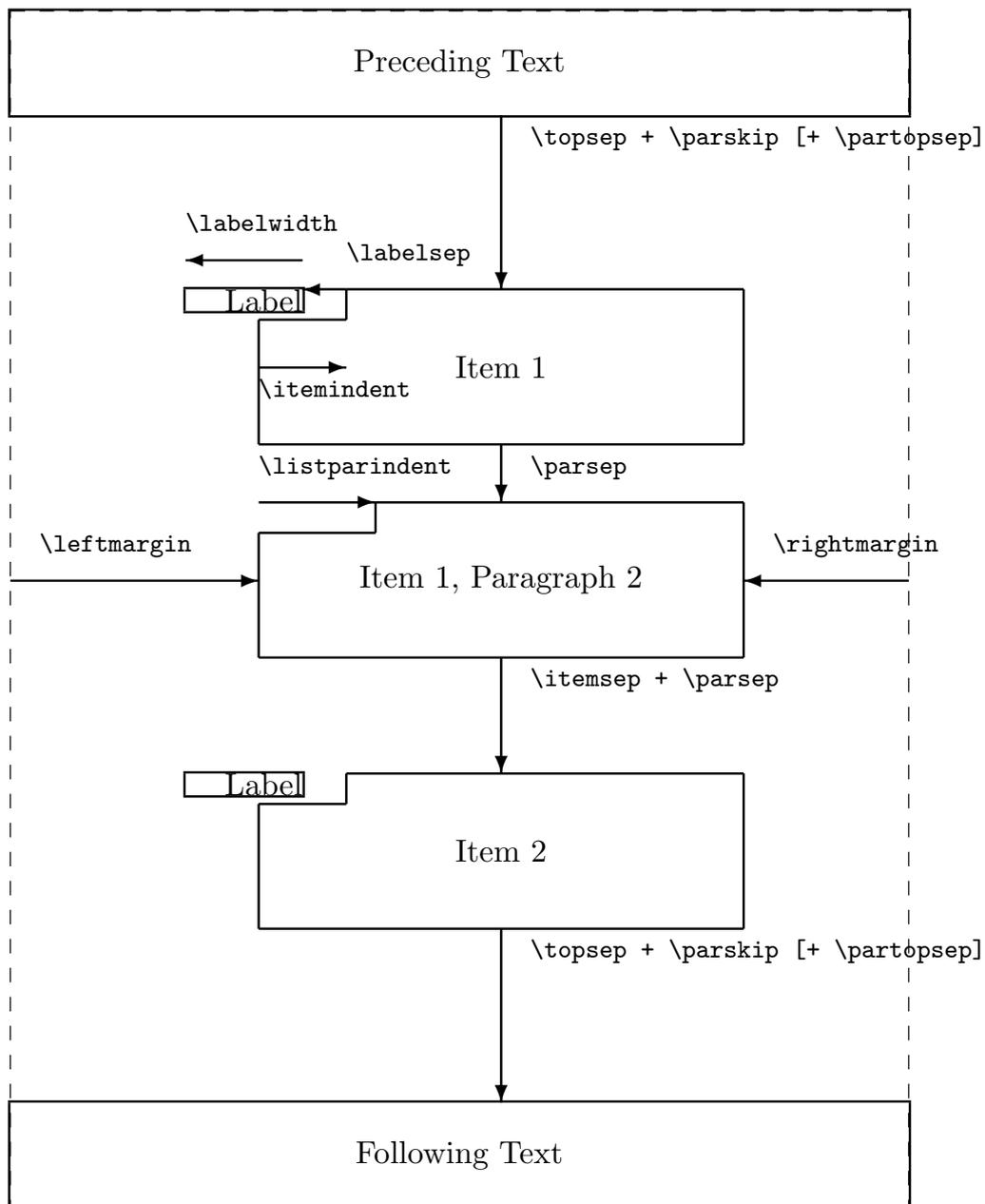
Der innere Aufbau einer Liste ist komplex (siehe Abb. 13.1, Seite 35). Deshalb gibt es Pakete, die bei der Veränderung der Listenformatierung helfen.

paralist

- ▶ definiert Listentypen `compactitem` und `compactenum` ohne vertikale Abstände (es können auch kleinere Abstände definiert werden, siehe Paketdokumentation);
- ▶ definiert Listentypen `inparaitem` und `inparaenum` für Aufzählungen in einem Absatz (auch kein Zeilenumbruch);
- ▶ definiert Listentypen `asparaitem`, `asparaenum` und `asparadesc` für Aufzählungen, bei denen jeder Listenpunkt wie ein gewöhnlicher Absatz formatiert ist;
- ▶ bietet Parameter [bzw. Schalter] `[&]pointedenum` und `[&]pointlessenum`, durch die z. B. die Labels einer zweistufigen Aufzählung als `1 1.1 , ...` bzw. `1 1.1 , ...` gesetzt werden;
- ▶ bietet Parameter `alwaysadjust`, der den für die Marken freigehaltenen Platz minimiert (falls dann die Einrückung nicht passt, die Aufzählungssymbole wie unten gezeigt explizit setzen);
- ▶ erlaubt eine einfache Umdefinition der Symbole für alle Arten von Listen durch einen optionalen Parameter:
`\begin{enumerate}[&{text}\formatbefehle <typ>]` wobei `<typ>` die Werte `1 i a A` annehmen kann
- ▶ und noch einiges mehr.

Abbildung 13.1: Struktur einer Liste mit den darin verwendeten Längen

Quelle: Dokumentation des Pakets layout

**hang**

- dient primär zum Setzen von Absätzen mit hängendem Einzug:

```
\setlength{\hangingindent}{length}
\begin{hangingpar}
```

```
...
```

```
\end{hangingpar}
```

- kann aber auch bequem einfache Listen erzeugen

```
\begin{labeledlist}{widest_label}
```

```
\item[label]...
```

```
\end{labeledlist} Neben labeledlist gibt es auch compactlabel für eine kompakte Liste.
```

easylist kann u. a. leicht “checklists” erzeugen:

<input type="checkbox"/> Languages	<code>\begin{easylist}[checklist]</code>
<input type="checkbox"/> LaTeX	<code># Languages</code>
<input type="checkbox"/> Fortran	<code>## LaTeX</code>
<input type="checkbox"/> Python	<code>## Fortran</code>
	<code>## Python</code>
	<code>\end{easylist}</code>

Man kann damit auch die Struktur einer umfangreicheren Arbeit entwerfen:

	<code>\begin{easylist}[articletoc]</code>
1 Introduction	<code># Introduction</code>
1.1 Motivation	<code>## Motivation</code>
1.2 Existing work	<code>## Existing work</code>
	<code># Data</code>
2 Data	<code>## Instruments</code>
2.1 Instruments	<code>### Instrument 1</code>
2.1.1 Instrument 1	<code>### Instrument 2</code>
2.1.2 Instrument 2	<code>\end{easylist}</code>

Andere Pakete:

Eine Übersicht über die zahlreichen anderen Pakete findet sich in

<https://www.ctan.org/topic/list>.

`enumitem` hat mit `paralist` überlappende Funktionalität, kann manchmal vorteilhaft sein.

14. Tabellen

14.1 Einfache Tabelle

Tabellen werden mit einer `\begin{tabular}[pos]{cols}...\end{tabular}` Umgebung erzeugt.

- ▶ Sie bilden nach außen eine einzige Box, können also nicht umgebrochen werden.
- ▶ Der `pos` Parameter wirkt wie bei `\parbox` (vertikale Position relativ zur selben Zeile), wird aber selten gebraucht, da Tabellen meist als eigener Absatz gesetzt werden.
- ▶ Die Tabelle ist nur so breit wie die Spalteninhalte es verlangen. Mit `tabular*` wird sie aber auf `\linewidth` ausgedehnt.
- ▶ Standardmäßig sind folgende Spaltentypen (`cols`) definiert:
 - linksbündig
 - centred
 - rechtsbündig
 - `paragraph`. In `p` und nur da ist ein Zeilenumbruch möglich; deshalb muss man `p` mit einer Breite angeben: `p{width}`. Ansonsten hat jede Zelle ihre natürliche Breite (maximaler Wert aller Zellen einer Spalte), die Tabellenbreite ergibt sich aus der Summe aller Spalten (+ Spaltenabstand).
 - Da der Befehl `\\` schon mit dem Ende der Tabellenzeile belegt ist (siehe unten), muss in einer `p`-Spalte ein allfälliger manueller Zeilenumbruch durch den Befehl `\newline` erzeugt werden.
 - `|` erzeugt eine vertikale Spaltentrennlinie.
 - `@{sep}` erzeugt einen expliziten Spaltentrenner. Implizit ist zwischen allen Spalten ein Leerraum `\hspace{\tabcolsep}=6.0pt` gesetzt. Zum Beispiel kann man mit `c@{:}c` zwei Spalten erzeugen, die durch `:` ohne Leerraum getrennt sind.
 - `*{N}{c}` Kurzschreibweise für `N` Spalten des Typs `c`
- ▶ Im Inneren der Tabellenumgebung gibt es folgende Kommandos:
 - `&` Spaltentrennzeichen
 - `\\` Ende der Zeile (wie üblich)
 - `\hline` horizontale Linie zwischen Zeilen
 - `\cline{i-j}` horizontale Line von Spalte `i` bis `j`. Keine normalen Zellen auf derselben Zeile!
- ▶ Weiters zu beachten:
 - Leerzeichen vor und nach `&` sowie am Anfang und Ende werden ignoriert. (NB: `TEXstudio` hebt die `&` nur hervor, wenn sie von einem Leer- oder Sonderzeichen gefolgt werden!)
 - Inhalt der Tabelle kann im Quellcode in beliebig viele Zeilen geteilt werden, aber keine leeren Zeilen! (Evt. mit `%` fast leere Zeile machen.)
 - `\setlength{\arrayrulewidth}{length}` Veränderung der Standardliniendicke von `\arrayrulewidth=0.4pt` (oder kurz `\arrayrulewidth length`)
 - Reguläre Fußnoten sind nicht möglich (und auch nicht üblich). Statt dessen mit z. B. `*` Pseudo-Fußnotensymbol erzeugen, unterhalb der Tabelle dann mit dem Pseudo-Fußnotentext wiederholen (oder ein Paket wie

Kurzbeschreibung
S. 28

Not So Short Intro
p. 34 – 37

<https://en.wikibooks.org/wiki/LaTeX/Tables>

verwenden, oder `longtable` [siehe Seite 40]; siehe auch <http://www.texfaq.org/FAQ-footintab>).

- TeXstudio stellt einen *wizard* “Quick tabular” zur Verfügung

Beispiel

Tag	Programm	Bemerkung
Montag	Vorlesung	fällt heute aus
Dienstag	Übung	17 Uhr

```

\begin{tabular}{l|cr}
\hline
Tag & Programm & Bemerkung \\
\hline
Montag & Vorlesung & fällt heute aus \\
Dienstag & Übung & 17 Uhr \\
\hline
\end{tabular}

```

14.2 Zellen, die über mehrere Spalten oder Zeilen gehen

Diese werden **horizontal** durch `\multicolumn{ncols}{coltype}{text}` gebildet, wobei `ncols` die Anzahl der zusammengefassten Spalten ist, `coltype` die Ausrichtung der neuen Spalte (wie oben, eins von `lrcp`, ggf. auch `|`), und `text` der Zelleninhalt.

Für eine **vertikale Zusammenfassung von Zellen** gibt es zwei Möglichkeiten:

1. Verschachtelte Tabellen. Hier kann man sich vorstellen, dass manche Zellen vertikal noch unterteilt sind. Die übergeordnete Tabelle enthält nur die nicht unterteilten Zeilen. Eine Zelle, die in n Zeilen zu teilen ist, wird durch eine eigene Tabelle mit n Zeilen ersetzt.

Hierbei kann es zu Problemen bei Verwendung von horizontalen Gitterlinien komme, die bei Option 2 vermieden werden.

2. Paket `multirow` mit der Syntax

`\multirow{nrows}{width}[fixup]{text}`

- ▷ `nrows` ist die Anzahl der *folgenden* Zeilen, die vereinigt werden sollen. Wenn < 0 , der *vorangehenden* Zeilen.
- ▷ `width` die Breite der jeweiligen Spalte (wirkt wie `p{width}`). In der Regel * angeben, d. h. natürliche Breite.
- ▷ `fixup` Optionale vertikale Verschiebung von `text`, da er evt. nicht vertikal zentriert erscheint.

Beispiel:

Regular line 1a	2a
Common text	2b
	2c
	2d
Regular line 1e	2e

```

\begin{tabular}{|c|c|}
\hline
Regular line 1a & 2a \\
\hline
\multirow{3}{*}{Common text} & 2b \\
& 2c \\
& 2d \\
\hline
Regular line 1e & 2e \\
\hline
\end{tabular}

```

14.3 Pakete zur Verbesserung und Erweiterung des Tabellen-Layouts

Es existieren zahlreiche solche Pakete (siehe <https://www.ctan.org/topic/table>). Empfohlen werden die im Folgenden genannten.

14.3.1 Paket array

“A new implementation of L^AT_EX’s tabular and array environment”

(NB. Die array-Umgebung wird im mathematischen Modus eingesetzt und ist der tabular-Umgebung gleichwertig.) Zusätzliche Funktionen:

- ▶ `\setlength{\extrarowheight}{length}` Die Standardzeilen haben zu wenig Abstand, wenn man sie durch horizontale Linien trennt. Dies kann mit diesem Befehl global vermieden werden (length z. B. 1pt). (Alternative `|\rule{0pt}{2ex}` oder dergleichen in jeder Zeile)
- ▶ Automatische Einfügung von Text (z. B. Formatierungsbefehlen) am Beginn und / oder Ende jeder Zelle einer Spalte in der Spaltendefinitionsliste mittels `>{text} c <{text}`, wobei c hier für eine beliebige Spaltenart steht.
- ▶ Neue Spaltenarten `\m{width}` und `\b{width}`, die wie p wirken, aber die vertikale Ausrichtung des Zelleninhalts ermöglichen, analog zu innerpos in `\parbox`:
 - m{} mittig
 - b{} (bottom) unten,
 - p{} (top) oben, ohne array einzig mögliche Ausrichtung
- ▶ Mehr `\tabcolsep` rund um |
- ▶ `!{x}` wobei x für eine Alternative zum vertikalen Strich steht (Unterschied zu @: `\tabcolsep` wird nicht verschluckt.)
- ▶ Makros für eigene Spaltendefinitionen:


```
\newcolumntype{x}{>{vorher}{spalte/n}<{nachher}}
```
- ▶ **Fazit: Verwenden!**

14.3.2 Pakete dcolumn, fcolumn, rccolumn

Das Paket **dcolumn** stellt am Dezimalzeichen ausgerichtete Spalten zur Verfügung. Lädt array automatisch. Syntax der neuen Spaltenart:

`D{d}{e}{[m.]n}`

- ▶ d Dezimaltrennzeichen, das im Quelltext verwendet wird (meist .)
- ▶ e Dezimaltrennzeichen, das in der Ausgabe verwendet wird (kann z. B. für deutsche Texte auf , gesetzt werden, oder auf `\cdot` [liefert z. B. 7·6])
- ▶ n Anzahl der Nachkommastellen, die bei Ausrichtung berücksichtigt werden
- ▶ m Optionaler Parameter mit Anzahl der Vorkommastellen (nur nötig, wenn man mit Ergebnis unzufrieden ist – wird sonst automatisch ermittelt)
- ▶ Die D-Spalten werden im mathematischen Modus gesetzt. Text, z. B. Einheiten oder Spaltenüberschriften, muss daher in `\multicolumn` gesetzt werden. Bei vielen Spalten empfiehlt sich eine Definition wie


```
\newcommand{\coltitle}[1]{\multicolumn{1}{c}{#1}}
```

Bei Verwendung des (empfohlenen) Pakets **siunitx** (siehe Kap. 20, S. 77) wird **dcolumn nicht gebraucht**. Dezimalspalten werden dann mit **S** gesetzt, Sonderbehandlung für Text beschränkt sich darauf, diesen in `\{ \}` einzuschließen.

Beispiel (im Code ist die S-Version auskommentiert enthalten)

D Beispiel	S Beispiel
1.0	1.0
1.000000	1.000 000
10000.00	10 000.00

```

\begin{tabular}{|D{.}{.}{3}|}
\mathrm{D}\multicolumn{1}{|c|}{Beispiel}\
% \begin{tabular}{|S|}
% {S}\ {Beispiel}\
\hline 1.0\
1.000000\
10000.00\
\end{tabular}

```

Die Pakete **fcolumn** und **rccol** erlauben zusätzliche Tausender-Kennzeichnung und Rundung, was das **siunitx**-Paket aber ebenfalls leistet. **fcolumn** kann zusätzlich Inhalte der **f**-Spalten addieren, was durch `\sumline` geschieht; **f**-Spalten sind für Währungen optimiert, allgemeiner ist das Format `F{Tausenderzeichen}{Dezimalzeichen}{Gruppenl"ange}{Nachkommastellen}` (Gruppenlänge ist auf 3 zu setzen, wenn man Tausendergruppen möchte).

14.3.3 Paket booktabs

Dieses Paket verbessert die horizontalen Linien und insbesondere die vertikalen Abstände vor und nach diesen Linien, so dass die im Buchsatz übliche Qualität ohne weiteres Zutun erreicht werden kann. Es lädt im Hintergrund auch `array`.

- ▷ Es definiert `\toprule`, `\midrule`, `\cmidrule{i-j}`, `\bottomrule` (`\cmidrule` entspricht `\cline`).
- ▷ Alle Rules haben einen optionalen Parameter `[linewidth]`.
- ▷ Wiederholen von Linien zur Erzeugung von Doppellinien etc. in mehreren Zeilen erfordert das Vorschalten von `\morecmidrules`

Beispiel (`\cmidrule` nur zur Illustration, eigentlich hier nicht nötig)

Tag	Programm	Dauer
		(h)
Montag	Vorlesung	2
Dienstag	Übung	1,5

```

\begin{tabular}{lcr}
\toprule
Tag & Programm & Dauer \\
\cmidrule{2-3}
& & (h) \\
\midrule
Montag & Vorlesung & 2 \\
Dienstag & "Übung & 1,5 \\
\bottomrule
\end{tabular}

```

Dieses Beispiel illustriert auch professionelle **Satzregeln für Tabellen**:

- ▷ *Keine vertikalen Linien* (Ausnahme: wenn es bei sehr vielen Spalten für die Lesbarkeit und Übersichtlichkeit hilfreich ist, können diese gruppiert werden)
- ▷ *Horizontale Linien* über und unter der Tabelle (können auch entfallen) sowie zwischen Kopf und Tabellenkörper, *aber nicht nach jeder Zeile!*

Fazit: Verwenden!

14.3.4 Paket longtable

Dieses Paket (oder das alternative `supertabular`, das wir hier nicht besprechen) ist erforderlich, um Tabellen zu setzen, die länger als eine Seite sind.

Syntax:

```

\begin{longtable}[hpos]{cols}
%
```

```

\caption{heading} % optional
<Head on first page> \endfirsthead
%
\caption[heading] % optional
<Head for following pages> \endhead % optional
%
<Foot of table> \endfoot
<Foot on last page> \endlastfoot % optional
%
<Table body>
%
\end{longtable}

```

Merkmale:

- Seitenumbruch ist sowohl automatisch (nach einer Zeile, aber nicht in ihr) oder explizit (`\newpage`) möglich. Umbruch kann (wie üblich) durch `*` oder `\nopagebreak` verhindert werden.
- Tabelle wird als Absatz, nicht als LR-Box gesetzt.
- Der Absatz kann mit `hpos` horizontal positioniert werden: **l, c, r**
- Es können normale Fußnotenbefehle verwendet werden.
- Es kann optional ein auf jeder Seite zu wiederholender Tabellenkopf oder -fuß gesetzt werden (Wenn `firsthead/lastfoot` nicht angegeben sind, wird der Kopf-/Fuß überall verwendet.)
- Es kann eine Tabellenüberschrift (caption) angegeben werden; mehr dazu bei den Gleitumgebungen (siehe Seite 57).
- Ansonsten funktioniert die `longtable` nahezu wie `tabular`.

Fazit: Verwenden, wenn a) Tabelle über mehr als eine Seite geht, oder b) reguläre Fußnoten in der Tabelle verlangt sind.

14.3.5 Farbige Zellen

Das empfohlene Farbpaket `xcolor` (siehe Kap. 22.1, S. 82) lädt mit der Option `table` das Paket `colortbl` und erweitert zugleich dessen Funktionalität. Damit stehen folgende Befehle zur Verfügung, mit denen (einzelne oder alle) Zellen, Zeilen, und Spalten eingefärbt werden können, sowie farbige Trennlinien ermöglicht werden:

- `\cellcolor{color}`
- `\rowcolor{color}`
- `\rowcolors{startrow}{color_odd}{color_even}` (alternierend)
- `\arrayrulecolor{color}`

Beispiel:

```

{\rowcolors{4}{white}{gray}
\arrayrulecolor{white}\arrayrulewidth1pt
\begin{tabular}{|>{\columncolor{yellow}}1|cc|}
\hline
\rowcolor{green}a & 1 & A \\ \hline
b & 2 & B \\
c & 3 & \cellcolor{blue}C \\ \hline
d & 4 & B \\
e & 5 & B \\
f & 6 & B \\ \hline
\end{tabular} }

```

a	1	A
b	2	B
c	3	C
d	4	B
e	5	B
f	6	B

14.4 Einbringen von Werten in eine tabular-Umgebung

Wenn die Werte mit einem *Programm* erzeugt werden, bietet es sich an, die Ausgabe gleich so zu formatieren, dass sie der Tabellenkörper-Syntax entspricht.

Kommen die Werte aus einer *Tabellenkalkulation*, sollten sie als *.csv*-Datei gespeichert werden (*comma-separated values*, `,` oder evt. – wenn `,` schon als Dezimalzeichen verwendet – alternativ `;` als Spaltentrennzeichen).

Dann gibt es zwei Möglichkeiten:

14.4.1 Tabellenkörper mit externem Skript erzeugen

Unter Linux ist das recht einfach:

```
sed "s/,/ \&/g" fname.csv | sed "s/./&\\\\/ " > fname.textbl1
```

übersetzt alle `,` in `&` und fügt am Ende jeder Zeile ein `\\` hinzu. Das Ergebnisfile `fname.textbl` kann entweder in den L^AT_EX-Quelltext hineinkopiert werden, oder mit `\input{fname.textbl}` an geeigneter Stelle automatisch eingefügt werden.

Wer mag, kann den *stream editor* `sed` auch unter Windows installieren:

<http://gnuwin32.sourceforge.net/packages/sed.htm> oder gleich

<https://cygwin.com/>

Man kann auch den Editor von `texstudio` verwenden:

- `csv`-Datei aus `texstudio` öffnen (Achtung, Alle Dateien muss als Filter für den Dateityp eingestellt sein)
- `<CTRL>R` (Suchen–Ersetzen): Zuerst die Spaltentrennzeichen von `,` oder (deutsche Lokalisierung) `;` auf `&` global ersetzen.
- Dann am Ende jeder Zeile `\\` einfügen. Das geht so:
 - > Schaltfläche `Reg` (regular expression) aktivieren,
 - > unter Suchen `(.*)` angeben (match full line and make it a group), und
 - > unter Ersetzen `\1\\` angeben (substitute back the group = full line, and add `\\`)
 - > Dann global ersetzen.

14.4.2 Tabellen aus der Statistiksoftware R

Man kann natürlich Resultate im `csv`-Format ausgeben lassen und dann so wie oben (bzw. unten mit `datatool`) weiter verarbeiten. Andererseits gibt es auch das R-Paket `xtable`, mit dem direkt L^AT_EX- (oder auch HTML-)Tabellen erzeugt werden können. Dazu dient die Funktion `print.xtable(...)`, die viele optionale Parameter hat, um die Details der Tabelle in L^AT_EX zu steuern. Siehe Dokumentation (<https://cran.r-project.org/web/packages/xtable/>).

14.4.3 Paketbündel datatool

Ein sehr mächtiges Bündel aus mehreren Paketen zur Verarbeitung von Spreadsheet-Inhalten, inklusive Erstellung einfacher Grafiken (da so etwas Rechenzeit kostet und in T_EX mühsam ist, bei größeren Datemengen lieber von externem Programm erstellen lassen.).

¹Dies gilt fuer die `tcsh`-Shell. Unter `bash` müssen beim `sed`-Befehl acht Backslashes angegeben werden, um die zwei gewünschten zu erzeugen.

Hier wird nur die Minimalsyntax vorgestellt:

Schritt 1: *database* dbname erstellen

`\DTLloaddb[options]{dbname}{fname.csv}` oder
`\DTLloadrawdb[options]{dbname}{filename.csv}`
 übersetzt Sonderzeichen automatisch in L^AT_EX-Syntax (ausgen. \)

options

legt die Spaltenbezeichnungen fest. Wenn keine Optionen angegeben sind, wird die erste Zeile als Spaltenbezeichnung (*key*) gelesen.

Beispiel:

`[noheader,keys={Col1,Col2,Col3,...},omitlines=n]`

definiert Spaltenbezeichnungen *Col1* etc.

Wenn **omitlines** angegeben ist, werden die ersten **n** Zeilen ignoriert.

Schritt 2: *database* dbname ausgeben

`\DTLdisplaydb[omit_cols]{dbname}` bzw.
`\DTLdisplaylongdb[omit_cols]{dbname}`

gibt die *database* dbname in einer in einer tabular- bzw. longtable-Umgebung aus. Tabellenkopf wird fett gedruckt.

omit_cols ist eine durch Komma getrennte Liste von Spalten(bezeichnungen), die dabei ausgelassen werden.

Wichtig wenn Computer deutsch lokalisiert ist:

In diesem Fall wird ein csv-File in der Regel so erstellt, dass ; das Spaltentrennzeichen ist, und , das Dezimalzeichen (Standard = Englisch: ; als Trennzeichen und . als Dezimalzeichen). Man kann zwar mittels `\DTLsetseparator{;}` das Trennzeichen vorgeben, und mittels `\DTLsetnumberchars{,}` das Komma als Dezimalzeichen, aber letzteres funktioniert offenbar nicht gut. Daher sollte der Dezimalpunkt verwendet bzw. das Komma durch einen Punkt ersetzt werden.

Für die Verwendung von Umlauten in UTF8-Form (also *nicht* als \ "a etc.) wird `\usepackage[utf8]{inputenc}`² und `\usepackage[T1]{fontenc}` gebraucht.

Schritt 2 (alternativ): Ausgabe mit expliziter Angabe der Tabelle

```
\begin{tabular}[pos]{cols}
\DTLforeach[condition]{dbname}% Iteration Start
{\keyA=Col1, \keyB=Col2}% Definition key-val-Liste
{\keyA & \keyB \\\} Tabellenzeile, über die iteriert wird
\end{tabular}
```

14.4.4 Übungsbeispiel Datatool

Schritt 1: *database* erstellen

`\DTLloaddb[options]{dbname}{fname.csv}`

options legt die Spaltenbezeichnungen fest. Wenn keine Optionen angegeben sind, wird jeweils die erste Zeile einer Spalte als Spaltenbezeichnung (*key*) gelesen.

Mit `[noheader,keys={Col1,Col2,Col3,...},omitlines=n]`

werden Spaltenbezeichnungen *Col1* etc. definiert.

²Ab T_EXLive 2018 ist diese Angabe entbehrlich, und UTF-8-Kodierung wird automatisch angenommen.

Wenn `omitlines=n` angegeben ist, werden die ersten `n` Zeilen ignoriert, ansonsten werden die Werte ab Zeile 1 verwendet, wobei der Inhalt von Zeile 1 als Spaltenüberschrift interpretiert wird.

Schritt 2: *database* ausgeben

```
\DTLdisplaydb[omit_cols]{dbname}
```

gibt die *database* `dbname` in einer in einer `tabular`- bzw. `longtable` (`\DTLdisplaylongdb`)-Umgebung aus. Tabellenkopf wird fett gedruckt. `omit_cols` ist eine durch Komma getrennte Liste von Spalten(bezeichnungen), die ausgelassen werden.

Schritt 2 (alternativ): Ausgabe mit expliziter Angabe der Tabelle

```
\begin{tabular}{cols} (normale Tabellendefinition einsetzen)
\DTLforeach[condition]{dbname}% Iteration Start
{\keyA=Col1, \keyB=Col2}% Definition key-val-Liste
{\keyA & \keyB \\\} Tabellenzeile, über die iteriert wird
\end{tabular}
```

`ColA` etc. muss den Spaltennamen entsprechen, die beim Laden des `csv`-Files entweder aus Zeile 1 oder explizit zugewiesen wurden. `\keyA` etc. kann beliebig benannt werden, muss aber ein Kommando sein (mit `\` beginnen, nur Buchstaben aber keine Ziffern im Namen!)

Schritt 3: Grafik erzeugen

```
\DTLplot{dbname}{x=X-Wert,y=Quadrat,style=lines,height=4cm,
xticpoints={-5,0,5},yticgap=5,xlabel=X-Werte,ylabel=Quadrat,box}
```

Mehrere Kurven in einer Grafik:

Fall 1: Jede Kurve hat ein File: siehe S. 150 Dokumentation

Fall 2: Kurven sind Spalten in einem File: komplizierter, siehe S. 158

Datool hat noch viele weitere Möglichkeiten, siehe Dokumentation.

Mit `height=` gibt man die Höhe an, man könnte statt dessen auch `width=` verwenden.

15. Abbildungen

15.1 Allgemeines zu Grafikdateiformaten

15.1.1 Grafikformate, die in \LaTeX verwendet werden können

Grafiken kann man

1. *in \LaTeX direkt erzeugen*, entweder in der \LaTeX -eigenen `picture`-Umgebung, oder mit den moderneren Werkzeugen wie `pstricks`, `pdftricks`, `TikZ` (datatool nutzt im Hintergrund `TikZ`).
2. *extern erzeugen und als File einbinden*. Folgende Grafikformate sind für externe Dateien (ohne Verwendung von Zusatzpaketen) zulässig:

pdflatex: pdf, jpg, png, (eps – wird intern automatisch in pdf umgewandelt)

latex: ps, eps

15.1.2 Vektor- und Rastergrafikformat

Es gibt zwei grundsätzlich verschiedene Arten, Abbildungen auf einem Rechner zu speichern:

- (a) als Ansammlung von *Bildelementen* („Vektorformat“)
- (b) als Matrix von *Bildpunkten* („Rasterformat“, „bitmap“)

Für wissenschaftliche Grafiken (die hauptsächlich aus Linien, Buchstaben, Zahlen, und evt. gefüllten Polygonen bestehen) ist das Vektorformat ideal, man kann sie aber natürlich auch in einem Rasterformat darstellen.

Digitale Fotos und ähnliche Bilder bestehen aus Rasterdaten.

15.1.3 Vektorformate

Das Vektorformat hat (wenn anwendbar) wesentliche Vorteile:

- meist viel kleinere Dateien
- ohne Qualitätsverluste beliebig skalierbar

Daher ist \LaTeX auf die Verwendung von Grafiken im Vektorformat eingerichtet. Wichtige Vektorformate sind

- **Postscript** (.ps) bzw. **Portable Document Format (PDF)** (.pdf).
Erzeugung in \LaTeX : `pstricks`, <http://tug.org/PSTricks>,
bzw. `pdftricks`, <http://www.ctan.org/pkg/pdftricks>.
- **Scalable vector graphics** (.svg): Relativ neues Format, insbesondere um Vektorgrafik auf Webseiten zu ermöglichen.
- \LaTeX -Picture-Umgebung und das Nachfolgeformat **Portable Graphics Format (PGF)** (.pgf).
Erzeugung in \LaTeX : `TikZ`, <http://sourceforge.net/projects/pgf/>

Mehr dazu in Abschnitt 15.2.

15.1.4 Rasterformate

Davon gibt es sehr viele, jedes Betriebssystem hat eigene, einige sind allgemein gebräuchlich, vor allem in Web-Anwendungen. Am bekanntesten: gif, jpeg (jpg), png, bmp, tiff (tif)

Grundlegend bei Rasterformaten ist die Auflösung, angegeben in dpi (dots per inch, Bildpunkte pro Zoll = 2.54 cm). Daher besteht folgender Zusammenhang zwischen Länge einer Bildseite, Auflösung und Bildgröße in Bildpunkten:

$$\begin{aligned} N_x &= m^{-1} r_x x; & N_y &= m^{-1} r_y y \\ x &= m N_x / r_x; & y &= m N_y / r_y \\ r_x &= m N_x / x; & r_y &= m N_y / y \end{aligned}$$

wobei x, y die Seitenlängen des Bildes (in cm oder in) sind, r_x, r_y die Auflösungen (in dpi) und $N_x \times N_y$ die Bildgröße (in Pixel).

Wenn man x, y in cm statt inch angeben möchte, ist der Maßstabsfaktor $m = 2.54$ bzw. $m^{-1} \approx 0.39370$ in/cm zu setzen (ansonsten $m = 1$).

Beispiel 1: Wie groß ist ein Bild mit 800×600 Bildpunkten auf einem Bildschirm mit 96 dpi Auflösung?

$$\begin{aligned} x &= m N_x / r_x = 2.54 \text{ cm/in} \cdot 800 / 96 \approx 21.2 \text{ cm} \\ y &= m N_y / r_y = 2.54 \text{ cm/in} \cdot 600 / 96 \approx 15.9 \text{ cm} \end{aligned}$$

Beispiel 2: Wieviele Bildpunkte enthält eine A4-Seite, die mit 300 dpi gescannt wurde?¹

$$N_x N_y = 2.54^{-2} \cdot 300^2 \cdot 21.0 \cdot 29.7 \approx 8\,700\,632$$

Damit ist die Filegröße für so eine Seite:

- ▷ Schwarzweiß-Liniengrafik (1 bit pro Pixel): ca. 8 Mbit = 1 MByte
- ▷ Schwarzweiß-Graustufen (8 bit pro Pixel): ca. 8 MByte
- ▷ Farbbild, 24 bit Farbtiefe (3×8 bit pro Pixel): ca. 24 MByte

8 bit Tiefe heisst, die Grundfarbe kann in $2^8 = 256$ Abstufungen vorkommen.

Aus diesen Zahlen ist ersichtlich, dass unkomprimierte Rasterdateien unvernünftig groß werden!! Es gibt im wesentlichen zwei Kompressionsarten:

Verlustfreie Kompression (GIF- und PNG-Format)

Hier werden wiederholte Folgen derselben Grau- bzw. Farbstufe in kompakter Form abgespeichert. Für typische wissenschaftliche Grafiken sehr effizient! (In sog. komprimierten TIFFs wird auch dieser Algorithmus verwendet, ist also ebenfalls verlustfrei.) Siehe http://de.wikipedia.org/wiki/Graphics_Interchange_Format. GIFs sind auf maximal 256 Farben pro Grafik limitiert.

Diese Limitation entfällt beim neueren Format PNG, das außerdem noch effizienter komprimiert und freie Software ist. Allerdings ist es von älterer Software nicht lesbar. Siehe http://de.wikipedia.org/wiki/Portable_Network_Graphics.

¹Hinweise zum Scannen siehe zum Beispiel <http://comment.univie.ac.at/00-2/11/>

Mehrere gif-Dateien können leicht zu Movies (Animationen) zusammengefasst werden. Ähnliche Funktionalität für PNG bieten MNG und APNG, sind aber nicht verbreitet. Ein gängiges Movie-Format für verlustbehaftete Kompression ist MPEG. (Da für Druckwerke nicht relevant, wird Videoerzeugung hier nicht behandelt.)

Auch die bekannten zip-Programme verschiedener Provenienz sind verlustfreie Komprimierer.

Verlustbehaftete Kompression (JPG-Format)

Die verlustfreie Bildkompression funktioniert für fotoähnliche Bilder schlecht bis gar nicht, da es hier i. a. keine wiederholten Pixelfolgen gibt. Hier wird fast immer jpeg eingesetzt, siehe <http://de.wikipedia.org/wiki/JPEG>. Je nach Kompressions- bzw. Qualitätsfaktor werden die Dateien immer kleiner und die Verluste größer. Ein Qualitätsfaktor (Werte von 0–1 möglich) von 80-95 % ist für viele Anwendungen noch optisch nahezu verlustfrei. Bei schlechter Auflösung wird geminderte Qualität früher sichtbar!

Verlustbehaftete Kompression darf nicht für Grafiken mit Linien oder Text (also die typischen wissenschaftlichen Abbildungen!) verwendet werden, liefert hässliche bis unlesbare Ergebnisse. Bei Angabe von 100 % als Qualitätsfaktor erfolgt auch im JPEG-Format eine verlustfreie Kompression, d. h. es wird immer noch komprimiert im Vergleich zu BMP (simplem bitmaps, früher Standard unter DOS und Windows) oder nichtkomprimierendem TIFF.

15.2 PS- und PDF-Format

Postscript ist eine Art Programmiersprache, mit der man Seiten beschreiben kann. Es wird ein Papierformat definiert, und dann werden Objekte auf einer solchen Seite angeordnet (neben Linien etc. auch Rastergrafiken!). Postscript ist von der Firma Adobe (1984), der Standard ist aber offen und es gibt freie Implementierungen für alle wichtigen Funktionen zum Handhaben von Postscript-Files. Postscript ist an sich reiner ASCII-Text ("Postscript Level 1"), in späteren Versionen ("Level 2 / 3") wurde aber eingeführt, dass der ASCII-Inhalt komprimiert werden kann (Dateigrößenreduktion bei sehr vielen (kleinen) Elementen wichtig); daher sind diese Files Binärdateien.

EPS (encapsulated postscript) enthält eine *bounding box*, die den (weißen) Rand rund um den eigentlichen Inhalt ausblendet, und optional auch ein Preview Image in einem Rasterformat. In der Regel pro Datei nur eine Seite.

PDF (Portable Document Format) basiert auf Postscript und ist im Wesentlichen eine Erweiterung. Adobe hat den Standard offen gelegt (erste Version 1993), baut aber laufend neue Funktionen ein. Seit 2008 auch ISO-Standard. Nützlich sind Bookmarks und interne Links (werden auch von \LaTeX unterstützt), die Einführung von Javascript und Multimediaobjekten hat das Format aber vulnerabel gemacht. Im Acrobat Reader (Programm von Adobe zum Darstellen von PDF-Dateien) werden laufend neue Sicherheitslücken gefunden und gepatched. Durch Deaktivieren der Js-Funktion kann man viele davon vermeiden. Die Linux-Version *acroread* erhält seit Version 9.5.5 nur mehr Sicherheitsupdates und ist auf manchen Systemen (Debian) umständlich zu installieren. Leider können freie PDF-Reader (in Linux standardmäßig installiert, in Windows z. B. SumatraPDFReader) weniger als *acroread*.

PDF und Postscript Dateien können leicht in einander umgewandelt werden (z.B. mit ghostscript, unter Linux auch mit ps2pdf bzw. pdf2ps – rufen intern gs auf).

Zu den verschiedenen PDF-Versionen siehe Kapitel 23.1, Seite 84.

15.3 Erzeugen von Vektor-(Postscript-)Grafiken

15.3.1 Externe Grafiken

Am besten ist es, wenn das Programm, mit dem die Grafik erzeugt wird, bereits Postscript oder PDF ausgeben kann. Bei den meisten traditionellen *Linux*-Programmen ist das der Fall. Hier werden meist sehr kleine Filegrößen erreicht. Einige Beispiele für solche Programme (nur freie Software):

- ▷ `xmgrace*` <http://plasma-gate.weizmann.ac.il/Grace/>
- ▷ `gnuplot*`
- ▷ `python matplotlib**` (requires Microsoft Visual C++ on Windows)
- ▷ `GMT**` (Generic Mapping Tool)
- ▷ `R**` statistics software
- ▷ `octave**` freier „Nachbau“ von Matlab

Unter *Windows* sind solche Programme weniger in Gebrauch, die mit * gekennzeichneten können zumindest mit Hilfe von Cygwin oder MinGW unter Windows verwendet werden, jene mit ** sogar direkt.

Für Software, die keinen (e)ps/pdf-Output anbietet, kann man einen generischen Postscript- oder PDF-Druckertreiber von Adobe installieren und damit in ein ps-/PDF-File drucken. Wenn das Programm Ausgabe in jpeg oder tiff erlaubt, kann man diese Files in Postscript/PDF konvertieren. Der Nachteil dabei ist, dass man nur eine Rasterdatei in einer Postscript-Hülle bekommt und die großen Vorteile (Skalierbarkeit, Dateigröße) verloren gehen. Professionelle Grafikprogramme wie Corel-Draw können auch mit Postscript umgehen bzw. Postscript erzeugen, allerdings kann auch dieses Programm aus einer Rasterdatei keine Vektordatei mehr machen (während das umgekehrte problemlos ist!).

Optionen, um in *Windows-Anwendungen eingebettete Grafiken* in eine Grafikdatei zu bekommen:

- ▷ Mittels copy-paste in ein ppt-File geben, und die ppt-Seite als jpeg (oder anderes Grafikformat) abspeichern.
- ▷ Neuere MS-Office Versionen können markierte Objekte als PDF abspeichern.
- ▷ Als `wmf` (windows meta file) oder `emf` (enhanced meta file) speichern und mit geeignetem Programm konvertieren (z. B. inkscape)

Fotoähnliche Grafiken sollten in einer Auflösung abgespeichert werden, die für die Endwiedergabe sinnvoll ist (d.h. ≤ 300 dpi im Endformat) und dann wenn nötig in Postscript/PDF konvertiert werden, z. B. mittels `gimp` oder (Windows) `irfanview` – aber Achtung, es bleibt eine Rastergrafik!

15.3.2 L^AT_EX-interne Grafiken

(L^A)T_EX bringt eine eigene `picture`-Umgebung mit. Darin definiert man eine Zeichenfläche samt Koordinatensystem mit beliebigen Einheiten. Dann folgen die Objekte. Das Ganze gibt ein L^AT_EX-Element, wie eine Tabelle. Eignung für alles, was man unter Windows z.B. mit der "Draw"-Funktion von Word oder Powerpoint

machen kann, sowie einfache mathematische Darstellungen. Händische Erstellung eher mühsam, aber es gibt das `xfig` Programm mit GUI.

Diese Umgebung wird heutzutage aber nur mehr wenig verwendet, da es wesentlich mächtigere Werkzeuge gibt, die allerdings externe Pakete brauchen.

Das Paket `pstricks` verwendet postscript primitives, macht diese aber durch zahlreiche Makros in \LaTeX -ähnlicher Syntax leichter verwendbar. Es braucht die Rechenfähigkeiten der Postscript-Programmiersprache und ist daher nur mit `latex`, nicht aber mit `pdflatex` verwendbar. Dafür gibt es das Paket `pdftricks`, das diese Einschränkung umgeht und so die `pstricks` wieder zugänglich macht, wenn auch mit etwas Zusatzaufwand. Wie das geht, steht in der Paketdokumentation `pdftricks/manual.pdf`. Alternativ kann man auch `pdftricks2` verwenden, das den ps-code im Hintergrund in ein eigenes File schreibt, dieses mit `latex` übersetzt und dann den Output in pdf konvertiert und einbindet.

Für `pdf \LaTeX` hat sich vorwiegend `TikZ/PGF` durchgesetzt. Dies ist so wie `pdftricks` eine Sammlung von Makros, nur dass im Hintergrund nicht postscript, sondern PGF (portable graphics format) steht, das mit TikZ ein Bündel bildet und auch von \TeX interpretiert wird. Ähnlich wie mit `xfig` gibt es für TikZ auch GUIs und Editoren, die die Arbeit erleichtern können. Details zu TikZ weiter unten in Abschnitt 15.6.

15.4 Umwandeln von Grafikdateien

15.4.1 Raster-to-raster, vector-to-raster

- Kommandozeile (Linux, Windows): `convert` aus dem `imagemagick`-Paket (<http://www.imagemagick.org/>), oder `graphicsmagick`.
- Interaktiv: `gimp` (<http://www.gimp.org/>) oder (Windows) `irfanview`.
- Unter Linux gibt es oft auch noch viele spezielle Konvertierprogramme wie `tiff2ps`, `jpeg2ps` usw. Oft muss man mit den verschiedenen Programmen und ihren Optionen herumprobieren, bis man ein Optimum gefunden hat.

15.4.2 Raster-to-vector

Etwas besonderes sind Programme, die Rastergrafiken in echte Vektorgrafiken umwandeln. Diese sind eine Art Abwandlung von OCR-Programmen. Sie setzen gute Bildqualität und Auflösung voraus, und können dann recht brauchbare Ergebnisse erzielen. Die wichtigsten solchen Programme sind `potrace` (<http://potrace.sourceforge.net/>) und `autotrace` (<http://autotrace.sourceforge.net/>).

15.4.3 Vector-to-vector

- Interaktiv, mit editieren: `inkscape`
- Kommandozeile: `ghostscript` bzw. diverse Programme, die darauf zurückgreifen, inkl. `gv`
- PDF Tools, siehe Anhang "PDF-Handling" aus dem IMG-Wiki (Seite 94ff.)
`pdftk` Manipulieren von PDF Dokumenten (Seitenextraktion etc.)
- Programme, die mit TeXLive mitkommen:
`epspdf -b file.ps file.pdf` (-b makes the BB; operates on pdf files as well)
`pdf2ps file.pdf file.eps`
`pdfcrop file.pdf` automatically crops white space with a BB

- **pdfbox**, eine Java-Applikation (läuft auf allen Betriebssystemen) mit ähnlicher Funktionalität wie **gv** und **pdftk** (siehe Kap. 23.4, S. 86). Es ist für *Windows empfohlen*, da es derzeit das einzige Programm zu sein scheint, mit dem man die Cursor-Koordinaten auf einer PDF-Seite herauslesen kann, die man für den `bb`-Parameter bei `\includegraphics` (siehe unten) braucht. Aufruf von der Kommandozeile zu diesem Zweck:

```
java -jar <pdfbox-app.jar> PDFReader <file.pdf>
```

wobei für `<pdfbox-app.jar>` der komplette Pfadname zu diesem File einzusetzen ist, und für `<file.pdf>` der Name des zu betrachtenden Files.

15.5 Einbinden von Grafiken in Latex

15.5.1 `graphicx`-Paket

Paket `\usepackage{graphicx}`

Typische Anwendung (Annahme: file enthält Bounding Box, z. B. durch `epspdf`, oder man möchte die ganze Seite):

```
\includegraphics[width=\textwidth]{file.pdf}
```

Wenn man ein Teilgebiet ausschneiden möchte:

```
\includegraphics[width=\textwidth, bb = x1 y1 x2 y2, clip]{file.pdf}
```

Nützlich zum Beispiel, wenn mehrere gleichartige Grafiken in einer Abbildung – man kann die Legende wegschneiden und nur einmal extra ausgeben.

Die Koordinaten findet man, indem man das einzubindende File mit `gv` oder `pdfbox` betrachtet – beim Bewegen des Cursors wird dessen Position in den benötigten Koordinaten angezeigt. Querformat (landscape) – Bilder muss man oft um 90 Grad drehen:

```
\includegraphics[height=\textwidth,angle=-90, bb=x2 y1 x1 y2, clip]{file.ps}
```

Optionen für `includegraphics`

Falls die Grafiken nicht im selben Verzeichnis liegen wie die `tex`-Files, ist der absolute oder relative Pfad mit dem Dateinamen anzugeben, oder man definiert einen bzw. mehrere Suchpfade wie unten beschrieben. Die Pfad- und Filenamen unterliegen gewissen Einschränkungen. Unter Linux sind z. B. Leerzeichen nicht erlaubt. Abhilfe kann das Paket `grffile` verschaffen.

Nun die wichtigsten Optionen (aus der Paketdokumentation `grfguide.pdf`, editiert):

bb siehe oben

clip Clip the graphic to the bounding box.

hiresbb Boolean valued key. If set to `true` (just specifying `hiresbb` is equivalent to `hiresbb=true`) then T_EX will look for `%HiResBoundingBox` lines rather than `%BoundingBox`

viewport The `viewport` key takes four arguments, just like `bb`. However in this case the values are taken relative to the origin specified by the bounding box in the file. So to ‘view’ the 1in square in the bottom left hand corner of the area specified by the bounding box, use the argument `viewport=0 0 72 72`.

trim Similar to `viewport`, but here the four lengths specify the amount to remove or add to each side. `trim= 1 2 3 4` ‘crops’ the picture by 1bp at the left, 2bp at the bottom, 3bp on the right and 4bp at the top.

angle Rotation angle in degree.

origin for rotation. See the documentation of `\rotatebox`.

width give any valid L^AT_EX length

height give any valid L^AT_EX length. Alternative to width. If both specified result may be distorted.

totalheight Specify the total height (height + depth) of the figure. This will differ from the 'height' if rotation has occurred. In particular if the figure has been rotated by -90° then it will have zero height but large depth.

keepaspectratio If set to true then specifying both 'width' and 'height' (or 'totalheight') does not distort the figure but scales such that neither of the specified dimensions is exceeded.

scale Scale factor.

draft Locally switches to draft mode.

page Includes a specific page from a multi-page PDF file. (Alternative: mit `pdftk` Seite extrahieren, ggf. mit `pdftk input.pdf burst` zerlegen, sowie Paket `pdfpages`).

Grafikfiles müssen da stehen, wo auch das tex-File steht, oder sie müssen mit dem entsprechenden Pfad angegeben werden. Eine Datei sollte aber nur einmal in allen Pfaden vorkommen! Stehen alle Grafiken anderswo, kann man dies vorgeben:

`\graphicspath{{dir1/}{dir2/}}` would cause the system to look in the subdirectories `eps` and `tiff` of the current directory. This is unix syntax (Anm.: angeblich auch für Windows), on a Mac it would be:

`\graphicspath{{:dir1:}{:dir2:}}` Note the differing conventions, an initial `:` is needed on Macintosh systems to denote the current folder, whereas on unix an initial `/` would denote the top level 'root' directory.

15.5.2 Paket `lscap`

Stellt `\begin{landscape}... \end{landscape}` bereit.

Der Inhalt der Umgebung wird um 90° rotiert und bildet eine eigene Seite, wobei Kopf- und Fußzeile nicht mitrotiert werden.

Für Tabellen und Abbildungen, die nur im Querformat Platz haben.

15.5.3 Textumflossene Abbildungen und Tabellen

Pakete `wrapfig`, `cutwin` und andere (z. B. `picinpar`). Mit `wrapfig`:

```
\begin{wrapfigure}[Zeilen]{pos}[Rand"uberhang]{width}
...
\end{wrapfigure}
```

- `Zeilen`: Anzahl der Zeilen, für die der Platz der Abbildung ausgespart werden soll. Automatisch berechnet wenn nicht angegeben.
- `pos`: `left`, `right`; `inner`, `outer` (bei doppelseitigem Layout)
- `Rand"uberhang`: ermöglicht das Hineinragen in den Seitenrand (L^AT_EX-Länge angeben)
- `width`: Breite des reservierten Raums

15.5.4 Einfügen von svg-Grafiken

Paket `svg`

```
\includesvg[width=...,height=...,pre-cmds=...]{filename.svg}
```

`pre-cmds` sind L^AT_EX-Befehle, die logisch vor dem `\include` eingefügt werden und zum Beispiel dazu dienen können, die Schriftart innerhalb der SVG-Grafik zu verändern. Das Paket ruft im Hintergrund `inkscape` auf um den Text im L^AT_EX-Format zu extrahieren. Dieser wird in der Folge von L^AT_EX mit übersetzt, wodurch eine Vereinheitlichung der Fonts mit dem Rest des Dokuments erreicht wird.

15.6 Arbeiten mit tikz

Die Basis ist das Paket `pgf.sty`. Da man einige Einarbeitungszeit braucht und die Nutzung eher komplex ist, wird man viel mit Bibliotheken (`pgflibrary`, `tikzlibrary`) arbeiten, die vordefinierte Funktionen enthalten, sowie Beispiele aus den zahlreichen *examples* und *galleries* adaptieren.

Hier einige wichtige Links:

- ▶ Projekt-Homepage <http://sourceforge.net/projects/pgf/>
- ▶ Liste der relevanten Pakete: <http://ctan.org/topic/pgf-tikz>
- ▶ Manual: <http://mirrors.ctan.org/graphics/pgf/base/doc/pgfmanual.pdf>
- ▶ TikZ and PGF Resources (Überblick):
<http://www.texample.net/tikz/resources/>
- ▶ PGF and TikZ examples gallery: <http://www.texample.net/tikz/examples/>
- ▶ Editierhilfen: für Windows TikZedit (<http://www.tikzedit.org/>),
für Linux Paket `qtikz` bzw. (KDE) `ktikz`, siehe auch
<http://www.hackenberg.at/ktikz-editor-for-the-tikz-language>
- ▶ `tikzDevice` – TikZ output from R:
<https://texample.net/tikz/examples/tikzdevice-demo/>

15.7 Zusammenfassende Empfehlungen

15.7.1 Grafikformate

Wissenschaftliche Grafiken wenn irgendmöglich als **Vektorgrafik im PDF-Format** oder direkt in L^AT_EX. Nur im Ausnahmefall als PNG, oder (sehr detailreiche und deshalb übergroße Abbildung) als JPG (aber hohen Qualitätsfaktor wählen und Ergebnis unter vergrößerter Ansicht prüfen), dann optimal bei 16 cm Breite ca. 2500 Pixel in x-Richtung. *Wissenschaftliche Grafiken verlustbehaftet zu komprimieren ist ein Fehler, den man leider immer wieder sieht!*

Fotos und ähnliches als JPEG-Datei.

Auf 16 cm Breite 1200 bis 2500 Bildpunkte, Qualitätsfaktor 85%.

15.7.2 Freie Software

Zweck	Windows	MacOS	Linux
(Pre)View ps/PDF	acroread, <i>pdfbox</i> <i>ghostview</i> , acroread Foxit, <i>Sumatra</i> Preview <i>evince</i> , <i>okular</i> etc.		
Edit PDF	<i>pdftk</i> , (<i>pdfbox</i>)		
Preview Rastergrafik	<i>irfanview</i>	xnview	<i>gthumb</i> , <i>eom</i> etc.
Edit Rastergrafik	<i>irfanview</i>	<i>gimp</i> Preview	<i>imagemagick</i>
Edit Vektorgrafik	<i>inkscape</i>		

Anmerkung: acroread und foxit sind frei benutzbar, aber proprietäre Software. Preview ist Teil des MacOS. Xnview ist proprietäre Software, frei für nichtkommerzielle Anwendungen. Echte Open-Source-Software in *kursiv*.

Some Links:

- ▷ <https://www.sumatrapdfreader.org/>
- ▷ <https://www.pdflabs.com/tools/pdftk-the-pdf-toolkit/>
- ▷ <https://www.irfanview.com/>
- ▷ <https://www.gimp.org/>
- ▷ <https://inkscape.org/>
- ▷ <https://pdfbox.apache.org>

16. Gleitumgebungen (floats)

16.1 Gleitende Tabellen und Abbildungen mit Legende

Kurzbeschreibung
S. 38-40
Not So Short Intro
p. 39-41

In wissenschaftlichen Texten werden Abbildungen meist oben oder evt. unten auf einer Seite angeordnet. Sie sind niemals Teil des Fließtexts. Dies hat auch für den Seitenumbruch große Vorteile, da bei Abbildungen von 20-50% der Seitenhöhe sonst kein vernünftiger Umbruch erreicht werden kann (das Word-Problem!). Dasselbe gilt für Tabellen.

In L^AT_EX gibt es die Umgebungen `figure` und `table`, die als *Gleitumgebungen* bezeichnet werden, ihre Inhalte als *floats*. Diese werden automatisch positioniert. Verwendung:

```
\begin{figure}[tbhp!]           \begin{table}[tbhp!]
<figure>                       \caption{Tabellenlegende ..}
\caption{Abbildungslegende ..}  \label{tab:quadrat}
\label{fig:quadrat}             <table>
\end{figure}                   \end{table}
```

Das Aussehen ist aus Abb. 16.1 und Tab. 16.1 ersichtlich (die Tabelle wurde hier zentriert gesetzt).

- ▶ `\caption[list entry]{heading}` erzeugt eine Legende, die automatisch nummeriert wird. Die optionale Kurzform dient für das Abbildungs- bzw. Tabellenverzeichnis (wird erzeugt durch `\listoffigures`, `\listoftables`).
 - In einer umfangreichen Arbeit folgt meist auf das Inhaltsverzeichnis (`.toc`) zuerst das Tabellenverzeichnis (`.lot`), dann das Abbildungsverzeichnis (`.lof`).¹
- ▶ Bei Tabellen steht üblicherweise die Legende `\caption{}` oberhalb, bei Abbildungen unterhalb.
- ▶ `\label{key}` (optional) dient dazu, später mit `\ref{key}` eine Referenz auf die Abbildung bzw. Tabelle zu erzeugen. Der `\ref{fig:quadrat}`-Befehl wird durch

¹.toc, .lot, .lof sind die File-Extensions, unter denen diese Daten von L^AT_EX zwischengespeichert werden.

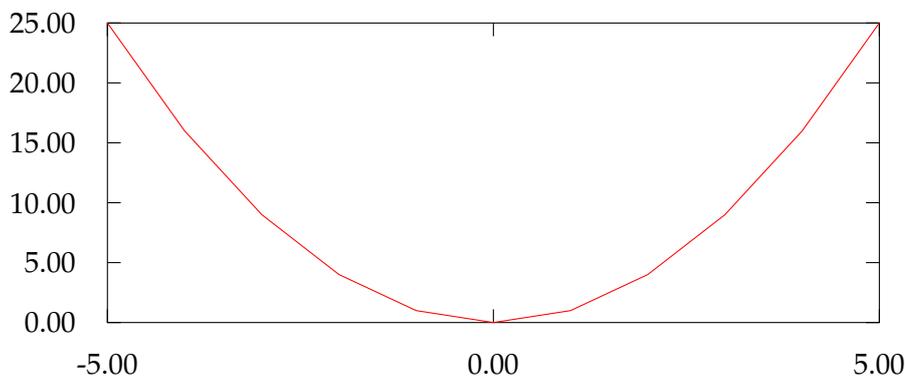


Abbildung 16.1: Eine Parabel zweiter Ordnung (blau nur als Demo).

die entsprechende Nummer ersetzt. Beispiel:

Das Aussehen ist aus Abb. 16.1 und Tab. 16.1 ersichtlich.

... aus Abb. `\ref{fig:quadrat}` und Tab. `\ref{tab:quadrat}` ...

Man kann das label-ref-System auch für andere nummerierte Objekte wie Gleichungen und Kapitel verwenden.

`\pageref{...}` liefert die Seitenzahl des jeweiligen labels und ist damit universell verwendbar. Bei Verwendung des Pakets `hyperref` werden solche Verweise in interne Links im PDF umgewandelt.

- Der optionale Parameter `[tbhp!]` (default: `tbp`) bestimmt, wohin das Float darf: `top`, `bottom`, `here`, `page of floats`. Mit `!` werden die u. g. Einschränkungen außer Kraft gesetzt). Wenn mehrere Optionen angegeben sind, haben `h` und `t` Priorität.
- Ein Float kann frühestens an der Stelle erscheinen, wo es definiert wird!
- \LaTeX hat Voreinstellungen, wieviel Platz Gleitobjekte auf einer Seite maximal einnehmen dürfen, die sehr restriktiv sind. Alles, was so nicht unterzubringen ist, wird auf Extraseiten am Schluss (*page of floats*) gesammelt. Um diese Einschränkungen aufzuheben bzw. zu mäßigen, kann man diverse Parameter umsetzen:

```

%% some modifications for placement of floats (figures, tables)
\renewcommand{\topfraction}{.95}      % max fraction of floats at top
\renewcommand{\bottomfraction}{0.8}   % max fraction of floats at bottom
\renewcommand{\textfraction}{0.0}     % minimum amount of text
                                      on a page with figs
%% Parameters for FLOAT pages (not text pages):
\renewcommand{\floatpagefraction}{.94} % require fuller float pages
    % N.B.: floatpagefraction MUST be less than topfraction !!

%% maximum number of floats at different positions:
\setcounter{topnumber}{3} % default: 2
\setcounter{bottomnumber}{1} % default: 1
\setcounter{totalnumber}{5} % default: 3

% for two-column pages (**does not apply to multicol**)
\renewcommand{\dbltopfraction}{.8} % vorher: .7
\renewcommand{\dblfloatpagefraction}{.6} % vorher: .5

```

Tabelle 16.1: Die tabellierten Werte der Parabel zweiter Ordnung.

X-Wert	Quadrat	Integriert
-5	25	25
-4	16	41
-3	9	50
-2	4	54
-1	1	55
0	0	55
1	1	56
2	4	60
3	9	69
4	16	85
5	25	110

- ▷ Das Ausgeben von *Floats* erzwingen (und zugleich eine neue Seite bzw. Doppelseite eröffnen):
`\clearpage` bzw. `\cleardoublepage`
 Das Erscheinen von (weiteren) *Floats* auf dieser Seite verhindern:
`\suppressfloats[tb]` (ggf. mit optionalem Parameter angeben, welche Positionierung betroffen sein soll)

16.2 Wichtiges Zusatzpaket: caption

`\usepackage{caption}`

`\captionsetup[figure/table]{...}`

erlaubt es, bequem das genaue Aussehen der Tabellenüberschriften und Abbildungsüberschriften zu bestimmen.

- ▷ `format=plain/hang`
- ▷ `labelformat=default/empty/simple/brace/parens/`
- ▷ `labelsep=none/colon/period/space/quad/newline/endash/`
- ▷ `justification=justified/centering/centerlast/centerfirst/raggedright/RaggedRight`
- ▷ `singlelinecheck=false/true`
- ▷ `font={...}, labelfont={...}, textfont={...}`
 Die Fontangaben können die üblichen Größenangaben (z. B. `small` – ohne vorangestellten `\!`) sowie die zwei-Buchstaben-Angaben zur Schriftformatierung wie `bf`, `it` etc. enthalten. Außerdem sind noch `color=<farbbefehl>` sowie die Zeilenabstandsangaben wie `singlespacing` etc. möglich.
- ▷ `margin`
- ▷ `position=bottom/top`, regelt Abstand caption – Inhalt. Für Tabelle wird automatisch `top` angenommen – ohne `caption`-Paket ist der Abstand `caption`–Tabelle zu gering!
- ▷ Für einige der Parameter sind auch selbstdefinierte Versionen möglich
- ▷ Dazu und noch einiges mehr, siehe Paketdokumentation!

16.3 Weitere Zusatzpakete

16.3.1 Paket chngcntr

In der `report`- und `book`-Klasse werden Tabellen, Abbildungen und Gleichungen kapitelweise nummeriert. Falls man das nicht möchte:

```
%% don't include chapter number in figures and tables and eqs
\usepackage{chngcntr}
\counterwithout{figure}{chapter}
\counterwithout{table}{chapter}
\counterwithout{equation}{chapter}
```

16.3.2 multicap

Für die Erzeugung von *captions* in `figure`- und `table`-Umgebungen bei Mehrspaltensatz mit dem Paket `multicol` (siehe Abschnitt 18.3, S. 70).

16.3.3 sidecap

Legende neben Abbildung oder Tabelle setzen.

```
\usepackage[option]{sidecap}
```

options: outercaption (default), innercaption, leftcaption, rightcaption, wide, ragged

```
\begin{Sctable}[relwidth][tbph!] ... \end{Sctable}
```

```
\begin{SCfigure}[relwidth][tbph!] ... \end{SCfigure}
```

16.3.4 floatrow

Normalerweise nehmen Gleitobjekte die gesamte Seitenbreite ein, auch wenn die eigentliche Abbildung oder Tabelle schmaler ist. Mit diesem, sehr umfangreichen Paket kann man unter anderem

- (1.) mehrere Gleitobjekte nebeneinander setzen, und
- (2.) die *caption* neben statt unter- oder oberhalb des *float*-Inhalts platzieren.

Dafür und die zahlreichen weiteren Optionen siehe Paketdokumentation. (Paket verändert diverse Einstellungen, nur verwenden wenn man es wirklich braucht.)

16.3.5 Tabelle oder Abbildung, die kein *float* sein soll

Falls man ausnahmsweise eine Abbildung oder Tabelle braucht, die nicht floaten soll, aber dennoch regulär nummeriert werden soll, gibt es folgende Optionen:

Paket **float**: Stellt einen Positionsparameter **H** zur Verfügung, der das *float* exakt an der jeweiligen Stelle einfügt. Wenn nicht genug Platz ist, wird eine neue Seite begonnen, auch wenn dann auf der vorherigen viel Leerraum bleibt.

Paket **nonfloat** gibt nur eine *caption* aus. Um diese mit dem Inhalt zusammen zu halten, muss man eine *minipage* verwenden. Beispiel (analog mit `\figcaption` für Abbildungen):

```
\\[\intextsep]
\begin{minipage}{\linewidth}
\tabcaption{Commands for Table and Figure Captions}%
\label{tab:Commands}%
\begin{tabular}{.....}...\end{tabular}
\end{minipage}
\\[\intextsep]
```

Das **caption**-Paket stellt einen Befehl

```
\captionof{floattype}[short heading]{heading}
```

zur Verfügung, der eine *Caption* auch außerhalb einer *float*-Umgebung erzeugt (aber innerhalb einer *minipage* oder anderen Umgebung). Für *floattype* ist dann *table* oder *figure* anzugeben.

longtable: Tabellen, die über mehr als eine Seite gehen, werden von der *float*-Umgebung nicht unterstützt. Jedoch bietet das *longtable*-Paket (das man dafür eh braucht – siehe Kap. 14.3.4, S. 40) die Option, im Abschnitt der Tabelle *vor* dem `\endhead`-Befehl einen `\caption{heading}`-Befehl anzugeben, so dass diese Tabelle regulär nummeriert wird (und auf den Folgeseiten ein entsprechender Hinweis am Anfang der Fortsetzungstabelle ausgegeben wird). Der `\label`-Befehl muss aber *nach* dem `\endhead` (oder innerhalb von `\firsthead`) kommen.

16.3.6 endfloat

Bei der Einreichung von Manuskripten für Zeitschriften wird oft verlangt, dass die Tabellen und Abbildungen gesammelt im Anschluss an den Textteil angeordnet werden. Oft ist auch eine Kennzeichnung der Stelle erwünscht, an der das Gleitobjekt im gedruckten Text in etwa stehen sollte. Dies leistet das Paket `endfloat`. Es kann durch eine Reihe von Optionen (der Paketdokumentation entnommen, siehe unten) konfiguriert werden (`tablesfirst` ist i. d. R. erwünscht).

Um mehr als ein float auf einer Seite zu platzieren, ist nach dem Laden des Pakets mittels

```
\renewcommand{\efloatseparator}{\relax}
```

der vordefinierte Trenner (standardmässig `\clearpage`) zu entfernen.

Liste der Optionen des Pakets `endfloat`:

Option	Default	Default implication	Descriptions
<code>nofiglist</code>	off		no list of figures
<code>notablist</code>	off		no list of tables
<code>nolists</code>		<code>nofiglist, notablist</code>	neither list
<code>figlist</code>	on		list of figures
<code>tablist</code>	on		list of tables
<code>lists</code>		<code>figlist, tablist</code>	list of tables and figures
<code>nofighead</code>	on		no 'Figures' section header
<code>notabhead</code>	on		no 'Tables' section header
<code>noheads</code>		<code>nofighead, notabhead</code>	neither of the headers
<code>fighead</code>	off		'Figures' section header
<code>tabhead</code>	off		'Tables' section header
<code>heads</code>		<code>fighead, tabhead</code>	Both section headers
<code>markers</code>	on		Place markers in the text
<code>nomarkers</code>	off		no markers in text
<code>tablesfirst</code>	off		Put tables before figures
<code>figuresfirst</code>	on		Put figures before tables
<code>tablesonly</code>			Disregard figures
<code>figuresonly</code>			Disregard tables

17. Literaturverweise und -verzeichnis

17.1 Einführung

Literaturverweise sind in einer wissenschaftlichen Arbeit essenziell. Sie werden ergänzt durch eine Quellenangabe für jeden Verweis. Die Quellenangabe kann entweder in einer Fußnote enthalten sein (in manchen Geisteswissenschaften üblich) oder in einem Literaturverzeichnis, das immer ganz am Schluss der Arbeit steht (nach Danksagungen, aber vor allfälligen Anhängen). Wir werden uns hier nur mit letzterer Version beschäftigen.

Die Verbindung von Verweis und Quelle wird durch einen so genannten `citekey` hergestellt, der frei wählbar ist. Günstig ist `AutorYYYYa` o. ä. (a, b, ... braucht man wenn es mehr als ein Werk pro Autor und Jahr gibt).

Es gibt drei Möglichkeiten, wie man arbeiten kann:

1. Händisch erstellte Umgebung

```
\begin{thebibliography}{widestlabel}...\end{thebibliography}
```

Darin müssen die einzelnen Quellen in folgender Form enthalten sein:

```
\bibitem[label]{citekey}Frei gestaltete Quellenangabe
```

Im Text sagt man `\cite{citekey1,citekey2,...}`. An dieser Stelle werden dann fortlaufende Nummern (in eckigen Klammern) ausgegeben, mit denen auch die Einträge im Literaturverzeichnis nummeriert werden. Wenn `label` angegeben ist, wird `label` an Stelle der Nummer verwendet. Die Labels stehen auch im Literaturverzeichnis, und die dafür reservierte Breite ist die von `widestlabel`.

2. Mit einer **Literaturdatenbank** und dem *Programm* **bibtex**, und ggf. Zusatzpaketen wie **natbib**

3. Mit einer **Literaturdatenbank** und dem *Paket* **biblatex** zusammen mit einem sog. *backend*, d. h.

(a) dem *Programm* **biblatex**, das im Hintergrund zusätzlich **bibtex** aufruft, oder

(b) dem *Programm* **biber**

Option 1 ist nur in Ausnahmefällen (ganz wenig Zitate) vernünftig.

Option 2 ist der traditionelle Standard, mit dem wir uns in der Folge näher beschäftigen. In den Naturwissenschaften ist die gängige Zitierweise im Text der sog. Author-Year-Stil: *Autor (Jahr)* bzw. *(Autor, Jahr)*. Diese wird am besten mit dem Paket **natbib** umgesetzt.

Option 3(a) ist eine Übergangsversion, Option 3(b) ist die modernere Nachfolgelösung von 2 und auch 3(a), die wir aus Zeitgründen nicht besprechen können. Da noch nicht alle Zeitschriften *styles* für **biblatex** zur Verfügung stellen, muss man sich mit 2 auf jeden Fall auskennen, und es reicht auch für die meisten Bedürfnisse völlig aus.

17.2 Literatur mit `bibtex` und `natbib`

Für Option 2 brauchen wir Folgendes:

1. eine Literaturdatenbank im `bibtex`-Format (`.bib`)
2. eine Literatur-Stildatei (`.bst`)
3. für Autor-Jahr-Zitierstil in der Regel auch das Paket `natbib`

Für viele Zeitschriften findet man die `bst`-Datei im Web (z. B. CTAN), oder sie wird vom Verlag zur Verfügung gestellt. Ansonsten verwendet man (mit `natbib`) die Standardstile `plainnat.bst` oder `abbrvnat.bst`. Man kann auch selbst einen neuen Stil mit Hilfe der Utility `makebst` produzieren (s. u.).

Ablauf beim Kompilieren:

1. `diss.tex` $\xrightarrow{\text{pdflatex}}$ `diss.pdf`, `diss.aux`, ...
 2. `diss.aux`, `literature.bib`, `dissref.bst` $\xrightarrow{\text{bibtex}}$ `diss.bbl`, `diss.blg`
 3. `diss.tex`, `diss.aux`, `diss.bbl` $\xrightarrow{\text{pdflatex}}$ `diss.pdf`, ...
- `.bbl`... bibliography ready to be included, `.blg`... bibtex log file

Arbeitsablauf:

1. Literaturdatenbank erstellen, ggf. während des Schreibens des Papers / der Arbeit
2. Die Literatur im Verlauf der Arbeit zitieren
3. Ein Literaturverzeichnis am Ende der Arbeit ausgeben lassen

17.2.1 Erstellen einer `bibtex`-Literaturdatenbank

`bibtex` erwartet die Literaturdatenbank in einem spezifischen `bibtex`-Format. Aus dieser holt es dann die angeforderten Einträge heraus, formatiert und sortiert sie (Schritt 2 oben).

Die `.bib`-Literaturdatenbank-Dateien sind reine ASCII-Files, die man mit jedem Texteditor (auch in TeXstudio) erstellen und bearbeiten kann. Wesentlich komfortabler und weniger fehleranfällig ist aber ein spezieller `bibtex`-Editor wie `jabref`. Das Arbeiten damit ist weitgehend selbsterklärend. Einige Grundbegriffe:

- ▶ Wie jede Datenbank, hat auch eine Literaturdatenbank *Felder*. Ein `bibtex`-File muss für jeden Eintrag bestimmte vordefinierte Felder enthalten, weitere frei definierbare Felder dürfen vorhanden sein (ob sie ausgegeben werden, bestimmt das `bst`-File). Typische Feldnamen sind `author`, `title`, `year`, `journal` etc.
- ▶ Es gibt definierte *entry types*. Jeder Eintrag muss einem dieser Typen zugeordnet werden, z. B. `book`, `article`, etc.
- ▶ Ein **Eintrag** sieht wie folgt aus


```
@entrytype{citekey,
  fieldname1 = {content1},
  ...
  fieldnameN = {contentN}
}
```

Wenn der Feldinhalt nur aus einem Wort besteht, braucht er nicht eingeklammert werden. Die Reihenfolge der Felder sowie der Einträge ist egal!

- **Entry types.** Es wird empfohlen, nur mit den folgenden Typen zu arbeiten:

article An article from a journal or magazine. *Required fields:* author, title, journal, year. *Optional fields:* volume, number, pages, month, note.

book A book with an explicit publisher. *Required fields:* author or editor, title, publisher, year. *Optional fields:* volume or number, series, address, edition, month, note.

inbook A part of a book, which may be a chapter (or section or whatever) and/or a range of pages. *Required fields:* author or editor, title, chapter and/or pages, publisher, year. *Optional fields:* volume or number, series, type, address, edition, month, note.

incollection A part of a book having its own title. *Required fields:* author, title, booktitle, publisher, year. *Optional fields:* editor, volume or number, series, type, chapter, pages, address, edition, month, note.

inproceedings An article in conference proceedings. *Required fields:* author, title, booktitle, year. *Optional fields:* editor, volume or number, series, pages, address, month, organization, publisher, note.

booklet A work that is printed and bound, but without a named publisher or sponsoring institution. *Required field:* title. *Optional fields:* author, howpublished, address, month, year, note.

misc Use this type when nothing else fits. *Required fields:* none. *Optional fields:* author, title, howpublished, month, year, note.

proceedings (kann durch **booklet** ersetzt werden) The proceedings of a conference. *Required fields:* title, year. *Optional fields:* editor, volume or number, series, address, month, organization, publisher, note.

Nichtempfohlen:

- **conference** (statt dessen **inproceedings**, sind ident)
- **manual, mastersthesis, phdthesis, techreport** (es wird ein Text wie *Ph. D. Thesis* oder *Tech. Report* im Literaturverzeichnis mit ausgegeben, den man oft nicht haben will; statt dessen für all diese „graue Literatur“ je nach Geschmack **booklet** oder **misc** verwenden)
- **unpublished** (kann in **misc** abgebildet werden)

- **Autorennamen.** Um verschiedene Bestandteile (Vorname(n), Nachname(n), Zusätze wie *van, de, jr.* etc.) richtig zuzuordnen, sowie die einzelnen Autoren voneinander zu trennen, muss eine definierte Syntax verwendet werden. Regeln dafür:

1. Mehrere Autoren sind durch **and** zu trennen.
2. Jeder einzelne Autorenname kann entweder in der Form **Vorname1 Vorname2 .. Nachname** oder **Nachname, Vorname1 Vorname2 ..** dargestellt werden. Vornamen können abgekürzt werden. Wenn nicht abgekürzt, so bestimmt das bst-File, ob im Literaturverzeichnis der volle oder der abgekürzte Vorname aufscheint.
3. Bei der Beistrich-Version wird alles vor dem Beistrich als Nachname, alles danach als Vorname aufgefasst. Wenn es Unklarheiten gibt, z. B. bei *Alexander Van der Bellen*, gruppieren:
Alexander {Van der Bellen}
4. Abgekürzte Vornamen immer mit **.** eingeben, den Punkt entfernen kann auch der Literaturstil!

5. Wenn eine Publikation keinen Autor und keinen Editor hat, kann man "Anonymous", ein Organisations-Acronym o. ä. angeben, damit diese besser zitierbar wird.
- ▶ **Zeitschriftennamen.** Die Crux liegt darin, dass diese entweder ausgeschrieben oder abgekürzt verwendet werden können. `jabref` hat eine eingebaute und ergänzbare lang-kurz Liste, mit deren Hilfe zwischen beiden Optionen umgeschaltet werden kann. Eine andere Möglichkeit ist es, im `bib-file` nur ganz kurze Abkürzungen zu verwenden, mit der syntax `journal = abbrev` (ohne `{}`)! In `Jabref` in `#. . . #`) und dann im `bst-file` die entsprechenden Definitionen als `MACRO {atmenv} {"Atmos. Environ."}` zur Verfügung zu stellen. Diese kann man bei Bedarf gegen eine Vollversion wie `MACRO {atmenv} {"Atmospheric Environment"}` austauschen.
 - ▶ **Titel.** Hier gilt es, auf Groß- oder Kleinschreibung zu achten. Im Englischen ist es üblich, dass in Buchtiteln Großschreibung (entsprechend der englischen Rechtschreibung!), in Artikel Titeln Kleinschreibung verwendet wird. `BibTEX` wandelt evt. im `title`-Feld eines `article` angegebene Großbuchstaben in Kleinbuchstaben um. Um diese zu schützen (Eigennamen, Titel in deutsch, etc.) muß man entweder den ganzen Eintrag oder Teile in ein zusätzliches `{}`-Paar einschachteln.
 - ▶ **DOI und URL.** Wenn man ein hinreichend neues `bst-File` hat, werden diese Felder unterstützt und sind natürlich sehr empfehlenswert. Oft ist aber in beiden Feldern dasselbe enthalten, da aus jedem DOI (digital object identifier) folgendermaßen ein URL (universal resource locator) konstruiert werden kann:
`doi:10.1000/demo_DOI` → `http://doi.org/10.1000/demo_DOI` .
 Damit man in PDFs anklickbare URLs hat (was im Übrigen das `hyperref`-Paket voraussetzt), ist die URL-Version zweckmäßig. Den DOI dann gar nicht extra im `bib-File` eintragen, oder im `bst-File` die DOI-Ausgabe unterdrücken.
 - ▶ **Umlaute u. a. nationale Sonderzeichen:** `BibTEX` versteht weder UTF-8 noch Abkürzungen wie "u! (Möglicherweise in neuesten Versionen anders – vorher probieren.) Umlaute etc. müssen daher mit `\"a` etc. geschrieben werden! Wenn es mit exotischen Sonderzeichen Probleme gibt, evt. zur Sicherheit noch in `{}`.
 - ▶ **Datenbankeinträge eintippen oder importieren?** Neue Literaturzitate kann man
 - selbst eintippen,
 - `jabref` aus einer Datenbank holen lassen (*Search / Web search*, oder DOI/ISBN im Feld *ID-based entry generator* [unten auf Pop-up-Fenster, welches sich mit der Schaltfläche `+` öffnet] eingeben)
 - sich z. B. auf der Webseite einer Zeitschrift den `bibtex-entry` ausgeben lassen, oder mit Suchmaschine danach suchen. Add-on <https://addons.mozilla.org/en-US/firefox/addon/jabfox/> kann dabei helfen. Aus der Zwischenablage einfach in die Literaturliste im Hauptfenster von `jabref` kopieren, neuer Eintrag wird automatisch angelegt; oft wird man ihn dann noch nachbearbeiten.
 - mit einem Syntax-Analyser halbautomatisch aus einem formatierten Zitat extrahieren lassen, in `jabref` mittels *BibT_EX / New entry from plain text*, oder mit dem Stand-alone-tool `cb2bib` (auch aus einer Titelseite oder `dgl.`), siehe <http://www.molspaces.com/cb2bib/>
 - und natürlich mit KollegInnen der Arbeitsgruppe teilen! `jabref` kann ab Version 4 statt `bib-Files` auch einen Datenbankserver nutzen, so dass alle Mitglieder der Gruppe eine gemeinsame Version auf dem neuesten Stand haben und auch bearbeiten können. Für die Verwendung in `BibTEX` auf ein `bib-File` augeben lassen: Tools → New sublibrary based on AUX file.

17.2.2 Zitieren mit natbib

Es gibt in bei Verwendung des Autor-Jahr-Systems im Wesentlichen zwei Zitierformen:

`\citep[postfix][prefix]{keylist}` citep – *in parenthesis*
`\citet[postfix][prefix]{keylist}` citet – *in text*

Die erste benötigt man, um einen Text wie *Wie der Literatur zu entnehmen ist (Someone, 1996; Somoneelse, 2009), ...* zu erzeugen, die zweite für *Wie bereits von Someone (1996) gezeigt, ...*

Die Keys in der Keylist werden mit `,` getrennt.

Typische Verwendung des optionalen `postfix` wäre eine Seiten- oder Kapitelangabe, von `prefix` zum Beispiel ein Vermerk *siehe*. Diese werden automatisch nach bzw. vor dem Zitat eingefügt, ggf. innerhalb der Klammer.

Mit `\nocite{*}` kann man den gesamten Inhalt des `bib`-Files ins Literaturverzeichnis übernehmen, auch wenn die Einträge nicht zitiert wurden; alternativ zu `*` kann man eine Liste von citekeys angeben.

Siehe Paketdokumentation für Spezialfälle und Modifikation des Zitierstils (numeriert, etc). Eigentlich sollten ohne besondere Angabe beim Zitat runde Klammern verwendet werden, falls doch eckige: `\usepackage[round]{natbib}`

In T_EXstudio werden (natürlich nur, wenn ein `bibfile` angegeenist, siehe unten) automatisch die verfügbaren citekeys angeboten, mit Zusatzinfo als Tooltip!

17.2.3 Ausgabe des Literaturverzeichnisses

Zuerst evt. Anpassungen vornehmen:

`\bibhang LENGTH` optional, hängender Einzug der bibitems
`\bibsep LENGTH` optional, vertikaler Abstand der bibitems
`\renewcommand{\bibname}{References}` (book, report) optional, statt *Bibliography*
`\renewcommand{\refname}{Bibliography}` (article) optional, statt *References*
`\phantomsection` nur für hyperref, um einen korrekten Link im toc zu erzeugen
`\addcontentsline{toc}{chapter}{\bibname}` optional, für Eintrag in toc; statt `chapter` bei Bedarf `section`!
`\raggedright` optional, ggf. für Anhang wieder aufheben durch `\justifying`

Dann Literaturverzeichnis erstellen:

`\bibliographystyle{stylefile}` bst-Filename ohne `.bst`
`\bibliography{bibfilelist}` ein oder mehrere `bib`-files, ohne `.bib`

In T_EXstudio sollte der grüne **Doppelpfeil** alle nötigen Compile-Läufe automatisch veranlassen. Wenn man z. B. zur Fehlersuche schrittweise vorgehen will:

1. `pdflatex diss[.tex]` erzeugt Informationen für 2. *Extension optional.*
2. `bibtex diss immer ohne .tex!` Produzieren der bibliografischen Informationen.
3. `pdflatex diss[.tex]` Einarbeiten der bibliografischen Informationen.
4. `pdflatex diss[.tex]` um alle Verzeichnisse, Bezüge etc. zu aktualisieren, evt. sogar $2\times$ oder gar $3\times$ – also insgesamt bis zu fünf oder sechs Kompilierschritte.

Schritt 2 braucht nur ausgeführt werden, wenn man ein neues bzw. korrigiertes Zitat oder einen neuen Zitierstil verwendet.

Das ausgegebene Literaturverzeichnis sollte sorgfältig korrektur gelesen werden – häufig hat man noch kleine Fehler im .bib-File, z. B. nicht geschützte Großbuchstaben, die in Kleinbuchstaben umgewandelt wurden.

Um einen eigenen Bibliographiestil zu erzeugen, kann man das Program `makebst` verwenden: `latex makebst` (Klommandozeile). Man muss dann alle Abfragen beantworten, und erhält am Schluss das neue .bst-File, auch wenn es etwas mühsam ist (wenn man sich bei einer Antwort vertippt, muss man von vorne anfangen.)

17.3 Konvertieren von Literaturdatenbanken

17.3.1 Hintergrund

An der BOKU und auch anderswo sind bei Nutzern von MS-Word bzw. Libre/Open-Office primär zwei Literatursoftwaresysteme üblich, nämlich das kommerzielle *Endnote*, und das freie *Zotero*.

Eine weitere Option zur Literaturdatenverwaltung, verbunden mit einer Art sozialem Netzwerk für WissenschaftlerInnen (mit all den Problematiken sozialer Netzwerke), ist das Webservice *Mendeley*, das aber u. a. deswegen in Verruf geraten ist, weil es vom Verlagsriesen Elsevier aufgekauft wurde, dem viele Universitätsbibliotheken überhöhte Zeitschriftenpreise vorwerfen.

Auch das Geschäftsgebaren von *Endnote* (gehört zum selben Konzern wie *Web of Science*, ein Quasi-Monopolist für die Sammlung von Daten über wissenschaftliche Zeitschriften und die darin erschienenen Artikel) ist fragwürdig – hohe Kosten für nicht-universitäre Lizenzen und eine 10-Millionen-Dollar-Klage gegen den Zotero-Filter für den Import aus Endnote.

17.3.2 Endnote

`jabref` kann verschiedene andere Literaturdatenbankformate einlesen. Da es ein proprietäres Format verwendet, ist *Endnote* nicht darunter (siehe das Schicksal von *Zotero*), aber dieses Programm kann selbst in `bibtex` exportieren (dann mit `jabref` importieren, `citekeys` erstellen, ggf. Umlaute, Großbuchstaben etc. nachbearbeiten).

Um eine .bib-Datenbank in *Endnote* zu bekommen, kann man diese von `jabref` in ein *Endnote*-kompatibles .txt-Format, oder in ein intermediäres Format wie RIS exportieren und dann importieren.

17.3.3 Zotero

Zotero kann in `bibTEX` exportieren, siehe http://libguides.mit.edu/ld.php?content_id=34248570. Angeblich kann man auch Einträge aus einem *Zotero*-Fenster in ein *Jabref*-Fenster „ziehen“.

Betreffend Austausch mit *Zotero* siehe auch <https://github.com/retorquere/zotero-better-bibtex>, damit kann eine *Zotero*- und eine `bibtex`-Datenbank synchronisiert werden.

17.3.4 Bib_TE_X in Office-Programmen verwenden

Es gibt auch noch weitere Optionen, um mit KollegInnen, die nicht mit L_AT_EX arbeiten, zumindest die bib_TE_X-Literaturdatenbank zu teilen, zum Beispiel

▷ Bibtex4Word (nur Windows):

http://www.ee.ic.ac.uk/hp/staff/dmb/perl/b4w_install.html

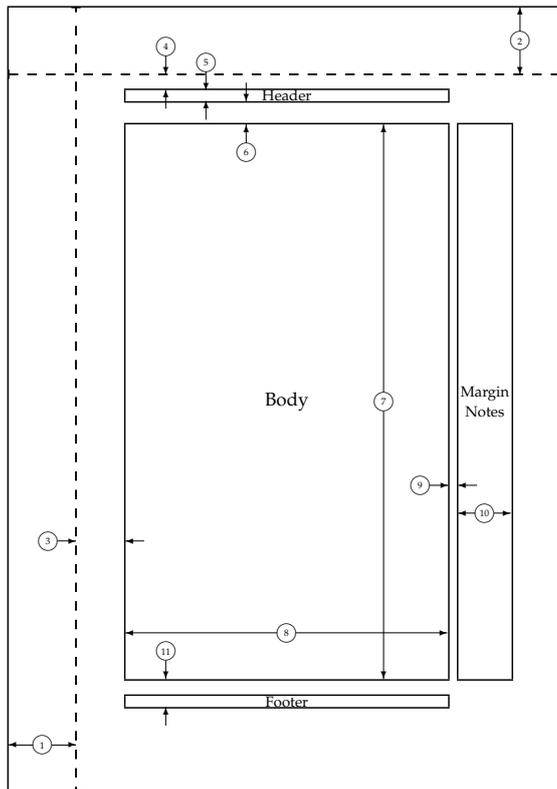
▷ jabref – OpenOffice/LibreOffice Integration:

<https://docs.jabref.org/cite/openofficeintegration>

18. Seitenlayout

18.1 Papierformat und Ränder mit geometry

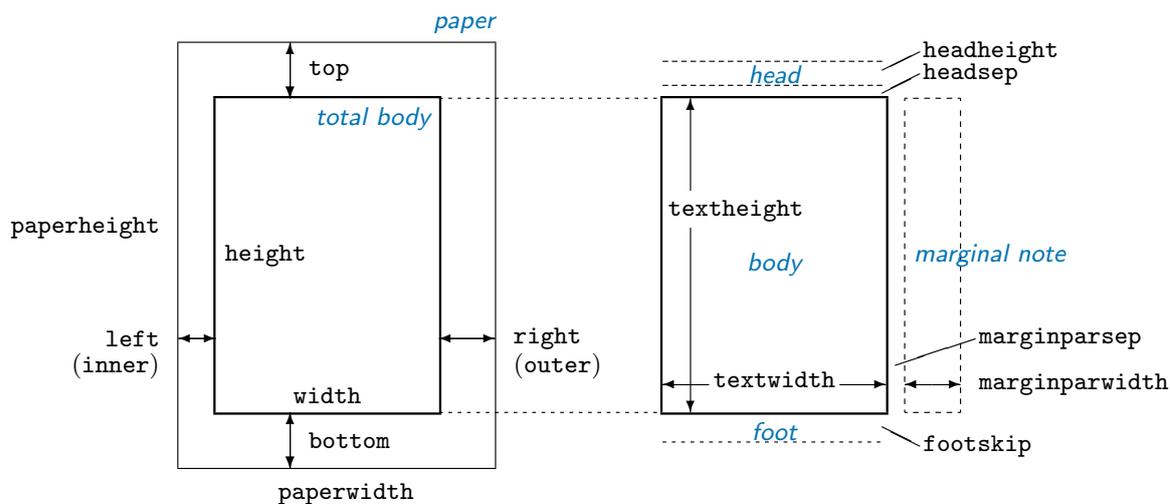
Das Standardlayout von L^AT_EX auf einer A4 Seite:



1	one inch + \hoffset	2	one inch + \voffset
3	\oddsidemargin = 53pt	4	\topmargin = 17pt
5	\headheight = 12pt	6	\headsep = 25pt
7	\textheight = 598pt	8	\textwidth = 345pt
9	\marginparsep = 11pt	10	\marginparwidth = 57pt
11	\footskip = 30pt		\marginparpush = 5pt (not shown)
	\hoffset = 0pt		\voffset = 0pt
	\paperwidth = 597pt		\paperheight = 845pt

erzeugt mit dem Paket layout

Die wichtigsten mit **geometry** (über Paketargumente) definierbaren Maße:



(a) default

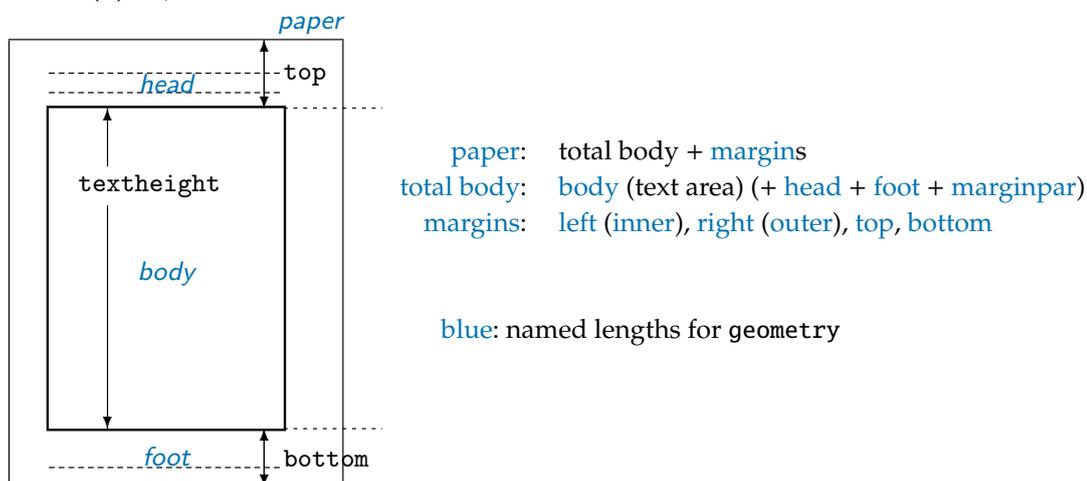


Abb. 2 (a) aus der Paketdokumentation; es wird empfohlen, die `includehead`-Version (Abb. 2 (b) in der Dokumentation, hier nicht gezeigt) *nicht* zu verwenden!

Papierformat

Entweder mit `a4paper` oder dergleichen angeben, oder mit `paperheight=...`, `paperwidth=...`

Default ist Hochformat (portrait), ggf. `landscape` angeben (aber nicht wenn schon mit `paperheight`, `paperwidth`!)

Mit `bindingoffset=...` kann man einen Rand (*Bundsteg*) definieren, der am *Innenrand* vom Papierformat abgezogen wird, wenn die Art der Bindung einen Teil des Papiers verschluckt.

Größe und Position des Textblocks

Kann entweder durch `height=...`, `width=...` oder implizit durch Angabe der Ränder definiert werden.

Randangabe: `margin=...` wenn allseitig gleich. Sonst entweder `top=`, `bottom=`, `left=`, `right=` oder `margin` und nur davon abweichende Randangaben.

Mit `headheight=` (ident: `head=`) und `headsep=` kann man Detailänderungen betr. Kopfzeilenraum machen. Der Fußraum ist `footskip` (ident: `foot=`).

Bei asymmetrischem Layout (für gerade=rechte und ungerade=linke Seiten) ist es wichtig, `twoside` als Hinweis auf den doppelseitigem Druck anzugeben.

Die Parameter `textheight`, `textwidth` sind die Maße des Textblocks im engeren Sinne (ohne Kopf und Fußzeilen und Randnotiz); sie sind auch verwendbare Variable (`\textheight`, `\textwidth`). Im hier gewählten Layoutmodell sind sie ident mit `height`, `width`.

`geometry` kann noch wesentlich mehr, wie Wechsel des Layouts im Dokument, siehe Dokumentation. Nützlich ist der Parameter `showframe`, der das Layout durch Linien zur Kontrolle hervorhebt.

18.2 Kopf- und Fußzeilen, `fancyhdr`

18.2.1 L^AT_EX-Standard

Der Inhalt von Kopf- und Fußzeilen ist in L^AT_EX für jede Dokumentenklasse vordefiniert. L^AT_EX hat vier standardisierte *Seitenstile* (`pagestyle`), d. h. vordefinierte Kopf- und Fußzeilen:

- ▷ `empty` weder Kopf noch Fuß
- ▷ `plain` nur Fußzeile, mit zentrierter Pagina (Seitenzahl)
- ▷ `headings` nur Kopfzeile, mit Kapitel- und Unterkapitelnummer und -name sowie Pagina.
- ▷ `myheadings` Kopfzeile, definierbar. Das Umdefinieren wird durch das Paket `fancyhdr` erleichtert und erweitert.

Der Seitenstil wird mit dem Schalter `\pagestyle{seitenstil}` verändert. Wenn man nur lokal auf einer Seite z. B. keine Paginierung (=Angabe der Seitenzahl) etc. möchte, erreicht man das durch `\thispagestyle{empty}`. L^AT_EX verwendet `\thispagestyle{plain}` in der Definition der chapter-Überschrift. Falls man das nicht möchte, kann man `plain` mit Hilfe von `\fancypagestyle{plain}{...}` (siehe unten) umdefinieren.

Der Befehl `\maketitle` setzt einen bestimmten Seitenstil (z. B. `plain` für die `article`-Dokumentenklasse); wenn man diesen verändern will, muss das `\pagestyle{option}`-Kommando daher *nach* `\maketitle` eingefügt werden.

18.2.2 Paket `fancyhdr`

Um Inhalt und/oder Formatierung zu verändern, verwendet man am besten das Paket `fancyhdr`. Es stellt einen neuen Seitenstil `fancy` zur Verfügung, der sowohl Kopf- wie Fußzeile beinhalten kann. In beiden kann man jeweils einen *linksbündigen*, einen *zentrierten* und einen *rechtsbündigen* Inhalt definieren.

Beispiel (aus der Paketdokumentation):

```
\pagestyle{fancy} % It is useful to do this first.
\fancyhf{} % clear all fields
\fancyhead{} % (oder z. B. nur:) clear all header fields
```

```

% set new header:
\fancyhead[RO,LE]{\bfseries The performance of new graduates}
% set new footer:
\fancyfoot{} % clear all footer fields
\fancyfoot[LE,RO]{\thepage}
\fancyfoot[LO,CE]{From: K. Grant}
\fancyfoot[CO,RE]{To: Dean A. Smith}
% change head- and footrules:
\renewcommand{\headrulewidth}{0.4pt}
\renewcommand{\footrulewidth}{0.4pt}

```

RO heißt right, odd (rechtsbündig für ungerade Seiten), **LE** steht für left, even, usw. **C** ist center. 0.4pt ist die Standardlinienstärke. Um Linie zu eliminieren, Stärke auf 0pt setzen.

Mit `\renewcommand{\headrule}{...}` kann man anstelle der normalerweise vorhandenen Linien beliebiges Material wie zum Beispiel Grafiken angeben, analog `\footrule`.

Mit `\fancypagestyle{name}{style}` kann man den Standard-Stil `name` umdefinieren, hier sind `style` alle anzuwendenden Definitionen wie oben. Als Abkürzung kann man mit `\fancypagestyle{plain}{}` für den `plain`-Stil die Definitionen von `fancy` einsetzen.

18.2.3 Anpassung der „lebenden Kolumnentitel“ (*running heads*)

(automatisch generierte Kopfzeilen etc. mit Kapitelinformation)

Details siehe `fancyhdr`-Paketdokumentation, Kapitel 9. Es gibt vier (bzw. fünf) relevante Befehle, die man umdefinieren kann (report funktioniert wie book):

	left (even) head	right (odd) head
Inhaltsdefinition:	<code>\markboth{lefthead}{righthead}</code>	<code>\markright{righthead}</code>
Formatierung book:	<code>\chaptermark{code}</code>	<code>\sectionmark{code}</code>
Formatierung article:	<code>\sectionmark{code}</code>	<code>\subsectionmark{code}</code>
Abruf über:	<code>\leftmark</code>	<code>\rightmark</code>

Bei `book`-Dokumenten ist das Argument `code` von

- ▷ `\chaptermark` das gesamte `\markboth{lefthead}{righthead}`, von
- ▷ `\sectionmark` das gesamte `\markright{righthead}`

und bei `article` ist das Argument von

- ▷ `\sectionmark` das gesamte `\markboth{lefthead}{righthead}`, von
- ▷ `\subsectionmark` das gesamte `\markright{righthead}`.

Wenn wir den Inhalt ändern wollen, zum Beispiel das Wort *Chapter* eliminieren, können wir schreiben

- ▷ `\renewcommand{\chaptermark}[1]{\markboth{\thechapter.\ #1}{}
\thechapter}` gibt die Kapitelnummer aus, und `#1` enthält den Kapitelnamen (genauer gesagt, die Kurzversion, falls sie definiert ist) – \LaTeX übergibt das, wenn das Kommando aufgerufen wird. `section`, `subsection` analog.

Ein Beispiel, wie die Kapitelinformation im `book`-Style weniger auffallend formatiert werden kann:

```

\fancyhf{} %remove all head and foot content
% put the page numbers in the outer corners (Right-Odd, Left-Even)
% and format them as small and bold:
\fancyhead[RO,LE]{\small\bfseries\thepage}
% now we redefine content and formatting:
\renewcommand{\chaptermark}[1]{\markboth{\thechapter.\ #1}{} }
\renewcommand{\sectionmark}[1]{\markright{\thesection.\ #1}}
% Also we put chapter info on the left and section info on the right page
% Formatting can be changed in the fancy-command as well.
\fancyhead[CO]{\small\rightmark}
\fancyhead[CE]{\small\leftmark}

```

Wenn nur die Großschreibung der `\rightmarks` weggebracht werden soll, reicht es, im fancy-command ein `\nouppercase` als Formatierung einzufügen.

18.2.4 Paginierung anpassen

Das fancyhdr-Paket hat auch einen Befehl, mit dem man die Paginierung von gewöhnlichen (arabischen) Ziffern auf römische (und umgekehrt) oder auch auf *a, b, c, ...* umstellen kann, und zwar mittels

```
\pagenumbering{arabic|roman|Roman|alph|Alph}
```

(einer der Werte ist anzugeben.) Gleichzeitig wird der Zähler (page) auf 1 zurückgesetzt – nützlich z. B. wenn ein Vorspann separat und mit römischen Ziffern paginiert werden soll.

18.3 Mehrspaltiger Satz mit multicol

```

\usepackage{multicol}
\begin{multicols}{ncol}[preface]
...
\end{multicols}

```

preface wird außerhalb der mehrzeiligen Umgebung gesetzt, z. B. für Überschrift oder Abstract.

Relevante Befehle und Parameter:

- ▶ `\columnbreak` Spaltenumbruch
- ▶ `\columnsep` Länge, die Spaltenabstand angibt
- ▶ `\columnseprule` Länge, die Breite einer Spaltentrennlinie angibt
- ▶ `\columnseprulecolor` Farbdefinition für Spaltentrennlinie

Das Paket versucht, die Spalten gleichmäßig zu füllen. Dagegen füllt die Version `\begin{multicols*}`... eine Spalte nach der anderen (auf jeder Seite).

18.4 Layoutempfehlungen

- Genug Rand lassen, Seiten nicht vollstopfen. Technische Texte nützen den Platz mehr aus, schöne Literatur ist großzügiger!
- Zeilen nicht zu lang (optimal 50–60 Z.), ggf. Zeilenabstand leicht vergrößern, aber nicht $1\frac{1}{2}$ - oder 2-zeilig, außer es will wirklich jemand Notizen zwischen den Zeilen machen.
- Mehr Raum unten als oben, mehr Raum außen als innen (manche sagen, außen = $2 \times$ innen, so dass bei aufgeschlagenem Buch der Abstand der beiden Textblöcke gleich wie der Randabstand ist). Professionelle Bücher als Vorbild nehmen.
- Seitenzahlen (*pagina*) entweder unten mittig, oder wenn eine Kopfzeile verwendet wird, in der Kopfzeile außen (außen sind sie beim Blättern besser zu finden). Unten außen ist auch möglich, aber nicht immer optisch ansprechend.
- Buchgestaltung ist ein Design-Beruf!

18.5 Formatierung von Kapitelüberschriften und Verzeichnissen

18.5.1 Festlegung, welche Überschriften ins Inhaltsverzeichnis kommen

Mit `\setcounter{tocdepth}{lev}` wird festgelegt, bis zu welcher Ebene (lev, z. B. 3) Überschriften ins Inhaltsverzeichnis aufgenommen werden (unabhängig von Paketen wie `titletoc`).

18.5.2 `titlesec`

Zur Veränderung der Darstellung der Kapitelüberschriften.

Einfache Verwendung indem eine oder mehrere der folgenden Paketoptionen angegeben werden (wirken global):

`rm sf tt`, `md bf`, `up it sl sc` Schriftart
`big medium small tiny` Schriftgröße relativ zu Standard
`raggedleft center raggedright` Ausrichtung
`compact` verringert Abstand vor und nach Überschrift
`uppercase` in Großbuchstaben

Die Formatierung der Kapitelnummern kann global angepasst werden mit

`\titlelabel{label-format}`

Beispiel: `\titlelabel{\thetitle.\quad}`

Die Formatierung der ganzen Überschrift kann pro Überschriftenebene verändert werden mit

`\titleformat*{command}{format}`

Beispiel: `\titleformat*{\section}{\itshape}`

Es kann aber auch jedes Detail festgelegt werden (komplizierter). Am besten, man geht von der Definition im Standard aus und verändert diese dann. Nur ein Beispiel (Kapitelüberschrift):

Die Kapitelüberschrift ist in der `titlesec`-Syntax standardmäßig so definiert:

```
\titleformat{\chapter}[display]
{\normalfont\huge\bfseries}{\chaptertitlename\ \thechapter}{20pt}{\Huge}
```

Wenn wir eine etwas kleinere Schrift wollen, alles in einer Zeile, und ohne das Wort „Kapitel“, dann können wir setzen:

```
\titleformat{\chapter}[hang]
{\normalfont\LARGE\bfseries}{\thechapter}{20pt}{\LARGE}
```

Die Abstände rund um den Überschriftsblock können so verändert werden:

```
\titlespacing{\chapter}{left}{before-sep}{after-sep}[right-sep]
```

Die Paketdokumentation enthält nicht nur alle Details, sondern auch einige Beispiele für stärker gestaltete Überschriften, die wir aber für eine Abschlussarbeit nicht unbedingt brauchen. Außerdem sind darin einige weitere Befehle beschrieben, die hier nicht diskutiert wurden.

18.5.3 `titletoc`

“Companion“-Paket von `titlesec` zur Anpassung der Inhaltsverzeichnisse, ebenfalls dokumentiert in `titlesec.pdf`. Die Option `dotinlabels` schließt im ToC die Kapitelnummern mit `.` ab. Festlegung sonstiger Details siehe Paketdokumentation.

18.6 Alternative Pakete

`titlesps`

Alternative zu `fancyhdr`, im Bündel mit den beiden anderen `title`-Paketen (`titletoc`, `titlesec`).

`tocloft`

Alternatives Paket für die Anpassung der Inhaltsverzeichnisse.

`tocbibind`

The `tocbibind` package enables the titles of the Table of Contents, the List of Figures, the List of Tables, the Bibliography and the Index all to be added to the Table of Contents. By default, all of these document elements, if they exist, will be incorporated into the Table of Contents (ToC for short). Package options are available to switch off any of these inclusions.

`etoc`

Völlig freie Gestaltung von Verzeichnissen, erfordert aber mehr Arbeit bei der Verwendung.

19. Satz mathematischer Formeln

Im mathematischen Modus

- werden die Satzregeln für mathematische Ausdrücke befolgt
- stehen zusätzliche Sonderzeichen zur Verfügung (mit den AMS-Paketen noch mehr)
- können Hoch- und Tiefstellungen (superscripts, subscripts) auf einfache Weise und mit automatischer Größenanpassung verwendet werden

Kurzbeschreibung
S. 29–36
Not So Short Intro
p. 43–72
Herbert Voß,
Mathematiksatz mit
L^AT_EX. 3. Aufl.

19.1 Arten von mathematischen Umgebungen

im Fließtext	<code>\$ \$</code>	<code>\(\)</code>	<code>\begin{math}</code>	<code>\end{math}</code>
nicht num. Gleichung	<code>\$\$ \$\$</code>	<code>\[\]</code>	<code>\begin{displaymath}</code>	<code>\end{displaymath}</code>
nummerierte Gleichung			<code>\begin{equation}</code>	<code>\end{equation}</code>
mehrzeilige num. Gleichung			<code>\begin{eqnarray}</code>	<code>\end{eqnarray}</code>
mehrzeilige nicht num. Gl.			<code>\begin{eqnarray*}</code>	<code>\end{eqnarray*}</code>

Empfohlene Alternativen für abgesetzte Gleichungen, setzt `amsmath`-Paket voraus:

nicht num. Gleichung			<code>\begin{equation*}</code>	<code>\end{equation*}</code>
nummerierte Gleichung			<code>\begin{equation}</code>	<code>\end{equation}</code>
mehrzeilige num. Gl.			<code>\begin{align}</code>	<code>\end{align}</code>
mehrzeilige nicht num. Gl.			<code>\begin{align*}</code>	<code>\end{align*}</code>
mehrzeilige Gl., eine Nr.			<code>\begin{equation}\begin{split}</code>	<code>\end{...}</code>

(jeweils) **empfohlen**

Vor und nach abgesetzten Gleichungen keine Leerzeilen lassen, wirken als Absätze! Ggf. Leerzeilen mit `%` zur optischen Gliederung des source codes.

Fließtext-Stil `\scriptstyle`, abgesetzt `\displaystyle`, beeinflusst Schriftgrößen bei Brüchen und wie Grenzen bei Integralen und Summen gesetzt werden.

19.2 Satzregeln und Satz-Beispiele

Skalare Variablen kursiv (ausgenommen griechische Großbuchstaben). Keine Wortabstände.

$xy + 1 = c. AB \alpha\beta \kappa\lambda \phi\varphi \Gamma\Delta - 2\pi$

`$x y +1=c. \; AB\; \alpha\beta\; \kappa\lambda\; \varphi\varphi\; \Gamma\Delta - 2\pi`
`\phi \varphi\; \Gamma\Delta - 2\pi`

Vektoren, Matrizen und Tensoren Vektoren kursiv, Matrizen und Tensoren meist Großbuchstaben und aufrecht oder serifenlos. Vektoren entweder mit Pfeil oder (Buchstabe) fett. Für fett und kursiv siehe unten AMSmath.

$\mathbf{v} \vec{v} \nabla \cdot \vec{v} \nabla \times \vec{v} \mathbf{P} \mathbf{P} \mathbf{P}$

`$ \mathbf{v} \quad \vec{v} \quad \nabla \cdot \vec{v} \quad \nabla \times \vec{v} \quad \mathbf{P} \quad \mathbf{P} \quad \mathbf{P}`
`\nabla \times \vec{v} \quad \mathbf{P} \quad \mathbf{P} \quad \mathbf{P}`

Allgemein definierte Funktionen Nicht kursiv!

$\sin \alpha \text{DIST}(x, y)$

`$ \sin\alpha \quad \text{DIST}(x, y)`

Horizontale Abstände Im *math mode* werden Leerzeichen verschluckt. Horizontale Abstände können erzeugt werden durch `\quad` `\qquad` `\sim` `\,` (wie im *text mode*), `\!` (kleiner negativer Abstand), `\>` (mittlerer Abstand) und `\;` (größerer Abstand).

Brüche $\frac{1}{2} \frac{a}{1-n}$ `\frac{1}{2}\;` `\displaystyle\frac{a}{\frac{1}{1-n}}\;`

Sub-/Superskripte, Exponent, Wurzel $\sqrt[3]{a^2} e^{-ax} k_1 a_{ijk}$
`\sqrt[3]{a^2} \;` `e^{-ax} \;` `k_1` `a_{ijk}\;`

Ableitungen, Integrale und Summen Auch Differentialsymbole wie *d* nicht als Variable = kursiv setzen!

$$\frac{df(x)}{dx} \frac{\partial f(x,y)}{\partial y} \int_{x_1}^{x_2} f(x) dx \sum_{i=1}^n (a_i - a_{i-1}) \iiint_V F dV \oint$$

`\[\frac{\mathrm{d}f(x)}{\mathrm{d}x} \;`
`\frac{\partial f(x,y)}{\partial y}`
`\int_{x_1}^{x_2} f(x)\,` `\mathrm{d}x` `\sum_{i=1}^n (a_i - a_{i-1})`
`\iiint\limits_V \!\!\! F \mathrm{d}V \oint \]`

Klammern L^AT_EX kennt verschiedene Arten von Klammern und klammerähnlichen Konstrukten.

- ▷ Allgemeine Struktur: `\left(... \right)` Größe *solcher* Klammern passt sich im Allgemeinen automatisch an
- ▷ Wichtigste mögliche Klammersymbole: `() [] \{ \} < > | |`
- ▷ wenn Anpassung nicht gut genug funktioniert, kann mit `\big` `\Big` `\bigg` `\Bigg` anstelle von `\left` und `\right` eine feste Größe eingestellt werden.
- ▷ Wenn Klammern nur einseitig verwendet werden, muss die fehlende Klammer durch ein Symbol wie `\left.` dargestellt sein.

array Umgebung für mehrspaltige Strukturen (Tensor, Matrix, aber auch Fallunterscheidungen etc.). Syntax wie `tabular`.

$$a(x,y) = \begin{cases} -1 & \text{für } x < 1 \\ 0 & \text{für } x = 0 \\ +1 & \text{für } x > 0 \wedge y \geq 0 \end{cases}$$

`\[a(x,y)=\left\{ \begin{array}{rcl} -1 & \& \mbox{f"ur}& x<1 \\ 0 & \& \mbox{f"ur}& x=0 \\ +1 & \& \mbox{f"ur}& x>0 \wedge y \geq 0 \end{array} \right. \]`

Schriftarten Im mathematischen Modus gibt es eigene Schriftfamilien-Kommandos:

`\mathbf{}` `\mathit{}` `\mathnormal{}` `\mathrm{}` `\mathsf{}` `\mathtt{}` `\mathcal{}`

Man kann sie aber nicht schachteln!

Beispiel für `\mathcal` (nur mit Großbuchstaben!) \mathcal{F}

Varianten für normalen Text im math-Modus (zur besseren Unterscheidbarkeit wird hier auf Sans-Serif-Schrift umgeschaltet):

`a` oderauch `b_{eff}` `\a\;` `\mathrm{oder auch}\;` `b_{\mathrm{eff}}\;`
`a` oderauch `b_{eff}` `\a\;` `\mathrm{oder auch}\;` `b_{\mathrm{eff}}\;`
`a` oder auch `b_{eff}` `\a\;` `\mbox{oder auch}\;` `b_{\mbox{eff}}\;`
`a` oder auch `b_{eff}` `\a\;` `\text{oder auch}\;` `b_{\text{eff}}\;` nur mit `amsmath`-Paket!

Not So Short Intro
p. 66 ff.

Es gibt zahlreiche mathematische Symbole und spezielle mathematische Akzente.

19.3 Fonts mit mathematischen Zeichen

Siehe dazu

- ▷ *A Survey of Free Math Fonts for TeX and LaTeX*
<http://www.tug.org/pracjourn/2006-1/hartke/hartke.pdf>
 (nicht mehr ganz aktuell aber immer noch nützlich)
- ▷ *LaTeX Font Catalogue*, Fonts with math:
<http://www.tug.dk/FontCatalogue/mathfonts.html>

Empfohlene Font-Pakete:

- ▷ Palatino: pxfonts bzw. \rightarrow
- ▷ Times: txfonts bzw. $\uparrow \rightarrow$
- ▷ Schoolbook: fouriernc
- ▷ Utopia: fourier
- ▷ Libertinus
- ▷ Standardfont: cm
- ▷ `\usepackage{newpxtext,newpxmath}` or `newtx...`
- ▷ `\usepackage[urw-garamond]{mathdesign}`
- ▷ `\usepackage{garamondx}`
- ▷ Präsentationen, Poster (sf): arev
- ▷ evt. `\usepackage[sfdefault,lining]{FiraSans}` (nur XeTeX)
- ▷ sf-Version von CM: cmbright

19.4 AMSmath und andere Zusatzpakete

Zusatzpakete für Mathematik-Satz allgemein:

<https://www.ctan.org/topic/math>

19.4.1 amsmath

`\usepackage{amsmath}` Dokumentation:

<http://mirrors.ctan.org/macros/latex/required/amslatex/math/amslatex/doc/amslatex.pdf>

amsmath vor anderen Paketen laden, sonst kann es zur Fehlermeldung

`\iint` already defined kommen.

amsmath sollte insbesondere verwendet werden, wenn man mehrzeilige Gleichungen verwenden will.

Wichtigste Features:

- ▷ Möglichkeit neue Funktionsnamen zu definieren
- ▷ Diverse Satzverbesserungen für (v. a. mehrzeilige) Gleichungen, insbesondere diverse align-Optionen
- ▷ `\eqref{label}` liefert (number)
- ▷ `*`-Form für alle Gleichungsumgebungen – nicht nummeriert
- ▷ Option `intlimits`: Integrationsgrenzen werden (im *displaystyle*) auch ohne Verwendung von `\int` `\limits` unter/über \int_0^1 statt als Sub/Superskripts \int_0^1 gesetzt.
- ▷ Umgebungen für Matrizen mit Klammern herum: `p|b|B|v|Vmatrix`

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad \text{\$}\begin{pmatrix} a & b \\ c & d \end{pmatrix}\text{\$}$$

$$\text{b: } [a] \quad \text{B: } \{a\} \quad \text{v: } |a| \quad \text{V: } \|a\|$$
- ▷ Mehr Abstandskommandos, inbes. `\mspace{mathlength}`, wobei `mathlength` in `mu=1/18em` anzugeben ist, z. B. `\mspace{-3mu}`
- ▷ Italic-Formen von griechischen Großbuchstaben wie Γ `\varGamma` statt Γ `\Gamma` etc.
- ▷ `\boldsymbol{argument}` für fette Versionen von Sonderzeichen und griechischen Buchstaben $\nabla\alpha$ statt $\nabla\alpha$, oder auch von kursiven Buchstaben:
 $A\mathit{v}$ `\$}\boldsymbol{\mathit{Av}}\text{\$}`; $A\mathbf{v}$
- ▷ `\text{text}` um normalen Text innerhalb des mathematischen Modus auszugeben, wobei anders als bei `\mbox{text}` die Größe angepasst wird (als Subskript `\scriptstyle`). Beachte dass Leerzeichen am Anfang und Ende des Arguments nicht unterdrückt werden. Nützlich zum Beispiel, wenn die `underbrace`-Funktion verwendet wird:

$$\underbrace{a^2}_I + \underbrace{b^2}_II = \underbrace{c^2}_{\text{Hypothenusenquadrat}}$$

`\underbrace{a^2}_{\text{I}} + \underbrace{b^2}_{\text{II}} = \underbrace{c^2}_{\text{Hypothenusenquadrat}}`

boldtensors

Bietet eine einfache Schreibweise für kursive, fette Buchstaben, wie sie oft für Vektoren und Tensoren verwendet wird: \vec{v} \vec{R} \vec{P} . Achtung, wenn `\mathbf{f}` verwendet wird, resultiert u. U. Fehlermeldung (“too many math alphabets”)! Daher Option `reuseMathAlphabets` verwenden.

isomath

Sorgt dafür, dass auch griechische Großbuchstaben automatisch kursiv gesetzt werden und stellt `\vectorsym{a}` `\matrixsym{A}` `\tensorsym{A}` aAA (vgl. aA regular math) bereit. Vektoren und Matrizennamen werden damit fett und kursiv gesetzt, Tensornamen zusätzlich in serifenloser Schrift. All das entspricht der ISO-Vorschrift für Mathematiksatz. Option `reuseMathAlphabets` wie oben.

mathfxs

Verbessert einige Kleinigkeiten, unter anderem:

- ▷ Stellt Befehl `\.` zur Verfügung, der kleinen Abstand erzeugt und Multiplikation symbolisieren soll.
- ▷ Stellt Befehl `\mathbold{}` für kursiv-fette Symbole zur Verfügung.

mismath

Enthält verschiedene neue Befehle, mit denen man leichter die Vorgaben für korrekten Satz erfüllen kann (z. B. `\di` für ein aufrechtes d als Differenzialsymbol, oder `\e` für aufrechtes e als Eulersche Zahl) und die kleine Verbesserungen bei Abständen und dergleichen erhalten kann.

20. Satz von physikalischen Einheiten mit SIunitx

20.1 Grundsätzliches zum Setzen von Einheiten

Ein Ausdruck wie **1 m** ist eine Abkürzung für **ein Meter**. Deshalb

- muss zwischen Zahl und Einheit ein Abstand sein, und
- darf die Einheit nicht im mathematischen Modus (kursiv) geschrieben werden, da z. B. *m* sonst die Bedeutung **Variable m** (z. B. Masse, oder was immer) hätte.
- Einheiten dürfen auch nicht in [] gesetzt werden.
(Manchmal wird [.] als Operator für die Dimension verwendet, das heisst [l] = m, wenn l eine Länge ist).

Ob man die Trennung von Wert und Einheit beim Zeilenumbruch verhindern möchte, ist Geschmackssache. Bei kurzen Einheiten wahrscheinlich wünschenswert, bei langen Zahl-Einheit-Ausdrücken kaum möglich.

Fünf Kubikmeter würde man so schreiben: 5 m^3 `5~m3` oder auch 5 m^3 `5\,m3` (kleinerer Abstand)

Wenn man exakt sein möchte, sollte man zwischen zwei Untereinheiten einen kleinen Zwischenraum lassen (kg m^{-3} `kg\,m--3`), aber das überlässt man besser dem Paket `siunitx`.

20.2 Verwendung von siunitx

Wenn man viele einheitenbehaftete Werte hat, ist das `siunitx`-Paket sehr hilfreich. Es kann (gleiche Syntax im Text- und Math-Modus)

- Zahlen formatieren `\num{number}`
- Dezimaltabulatoren in Tabellen mit Spaltentyp `S`
- Einheiten erzeugen mit `\si{unit}`
- Zahl mit Einheit formatiert ausgeben `\SI{value}{unit}`
- und noch einige weniger wichtige Funktionen ...

Empfohlene Anpassungen:

```
\usepackage{siunitx}
\sisetup{% changing defaults for siunitx
  exponent-product=!\cdot!, % otherwise would be $\times$
  input-decimal-markers={,} % only if you want to provide your numbers with ,
  output-decimal-marker={,} % only if you want decimal marker , in print
  number-unit-product = \ % if number-unit space shall be full space }
```

Beispiele

1 m^3	<code>\SI{1}{\metre\cubed}</code>
1 m s^{-1}	<code>\SI{1}{\metre\per\second}</code> aber
1 m/s	<code>\SI{1}{m\per s}</code>
$1.5 \cdot 10^5 \text{ kg m}^{-3}$	<code>\SI{1.5e5}{\kilo\gram\per\metre\cubed}</code>
$0.05 \mu\text{Sv h}^{-1}$	<code>\SI{0.05}{\micro\sievert\per\hour}</code>
$2500 \text{ J kg}^{-1} \text{ K}^{-1}$	<code>\SI{2500}{\joule\per\kilogram\per\kelvin}</code>
$-12 \text{ }^\circ\text{C}$	<code>\SI{-12}{\celsius}</code>
90°	<code>\SI{90}{\degree}</code>
0.4 m^4	<code>SI{0.4}{m\tothe{4}}</code>
$1 \mu\text{m}$	<code>\SI{1}{\um}</code> diverse Abkürzungen verfügbar!

Das Paket ist hoch konfigurierbar und bietet diverse weitere Optionen, siehe Dokumentation. Für öfter gebrauchte Einheiten empfiehlt sich eine Abkürzung mittels `\newcommand` oder auch `\DeclareSIUnit\kgmc{\si{kilo\gram\per\metre\cubed}}` womit man z. B. schreiben kann `\SI{1}{\kgmc}` und 1 kg m^{-3} erhält. Man könnte sich auch `\kgmc{number}` als noch kürzere Schreibweise definieren.

21. Satz von chemischen Formeln

21.1 Summenformeln mit mhchem

`\usepackage[version=4]{mhchem}`

Bei diesem Paket muss die Versionsnummer eingegeben werden, derzeit typischerweise Version 4. Wie kann die Version gefunden werden, die am eigenen Rechner gerade installiert ist?

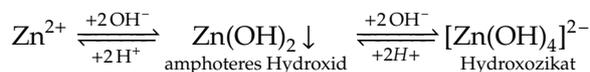
- Option 1: Paket einbinden, dann im Logfile nach einer Zeile suchen wie
`Package: mhchem 2018/06/22 v4.08 for typesetting chemical formulae`
 – hier also Version 4 (Subversion .08 ist nicht relevant)
- Option 2: Nach dem File `mhchem.sty` auf dem eigenen Rechner suchen, dort steht dann am Anfang z. B.
`\ProvidesPackage{mhchem}[2018/06/22 v4.08 for typesetting chemical formulae]`
- Option 3: Im lokalen Manual `mhchem.pdf` nachschauen, z. B. mittels `texdoc mhchem`, was dort als Versionsnummer steht

21.1.1 Verwendungsbeispiele

(z. T. aus der Dokumentation)

H_2O	<code>\ce{H2O}</code>
H^+	<code>\ce{H+}</code>
NO_3^-	<code>\ce{NO3-}</code>
$\frac{1}{2}\text{H}_2\text{O}$	<code>\ce{1/2H2O}</code>
${}_{90}^{227}\text{Th}^+$	<code>\ce{^227}_{90}Th+</code>
$\text{KCr}(\text{SO}_4)_2 \cdot 12\text{H}_2\text{O}$	<code>\ce{KCr(SO4)2 * 12H2O}</code>
Ce^{IV}	<code>\ce{Ce^{IV}}</code>
$\text{CO}_2 + \text{C} \longrightarrow 2\text{CO}$	<code>\ce{CO2 + C -> 2CO}</code>
$\text{CO}_2 + \text{C} \xleftarrow{\text{above}} 2\text{CO}$	<code>\ce{CO2 + C <-[\text{above}] 2CO}</code>
$\text{H}^+ + \text{OH}^- \rightleftharpoons \text{H}_2\text{O}$	<code>\ce{H+ + OH- <=> H2O}</code>
OCO^-	<code>\ce{OCO^{.-}}</code>
$\text{X}=\text{Y}$	<code>\ce{X=Y}</code>

`mhchem` stellt auch das `AMSMath`-Paket zur Verfügung und man kann mathematischen und chemischen Satz kombinieren:



`\ce{Zn^2+ <=>[\ce{+ 2OH-}][\ce{+ 2H+}]`
`\underset{\text{\text{amphoterer Hydroxid}}}{\ce{Zn(OH)2 v}}$`
`<=>C[+2OH-][+ 2H+]$ \underset{\text{\text{Hydroxozikat}}}{\ce{[Zn(OH)4]^2-}}$`

21.2 Sicherheitshinweise mit hpstatement und rsphrase

`mhchem` kommt im Bündel mit zwei weiteren Paketen:

`hpstatement` contains all official *hazard statements and precautionary statements (H and P) of the Globally Harmonized System of Classification and Labeling of Chemicals (GHS)* and of the *CLP Regulation of the European Union*. The statements are available in English and German.

`rsphrase` contains the text of all official *Risk and Safety (R and S) Phrases* that were used to label chemicals. These phrases are available in Danish, Englisch, French, German, Spanish, and Italian.

21.3 Strukturformeln mit chemfig

Strukturformeln sind naturgemäß wesentlich komplizierter. Eine Option sind externe Programme so wie für Grafiken, siehe z. B. Übersicht in https://en.wikipedia.org/wiki/Molecule_editor

Unter den verschiedenen Paketen für L^AT_EX hat sich `chemfig` inzwischen als gängigstes etabliert; es basiert auf TikZ. Andere Pakete sind PPCHTeX, XymTeX, OCHEM und chemstruct.

Anleitungen zu `chemfig`:

https://en.wikibooks.org/wiki/LaTeX/Chemical_Graphics sowie die als Tutorial gestaltete Paketdokumentation `chemfig_doc_en.pdf` (auf eigenem Rechner bzw. http://mirrors.ctan.org/macros/generic/chemfig/chemfig_doc_en.pdf)

21.3.1 Grundprinzip

```
\chemfig{ atom1 bondtype[angle,coeff,ndep,narr,tikzcode] atom2 }
```

Die "atoms" sind Atome oder auch Atomgruppen.

Sub- und Superskripts mit `^` `_` (ggf. hoch- oder tiefzustellenden Text in `{}`).

L^AT_EX-Schriftgrößenbefehle werden respektiert (*vor* dem `chemfig`-Befehl!)

Bindungsart

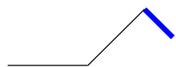
A — B	<code>\chemfig{A-B}</code>
A = B	<code>\chemfig{A=B}</code>
A ≡ B	<code>\chemfig{A~B}</code>
A ► B	<code>\chemfig{A>B}</code>
A ◄ B	<code>\chemfig{A<B}</code>
A B	<code>\chemfig{A>:B}</code>
A ··· B	<code>\chemfig{A<:B}</code>
A ▷ B	<code>\chemfig{A> B}</code>
A ◁ B	<code>\chemfig{A< B}</code>

Winkelangaben

- ▷ `[:absolute_angle]` $-360 \dots +360$, absolut, d. h. relativ zur Grundlinie
- ▷ `[:relative_angle]` $-360 \dots +360$, relativ zur vorherigen Bindungslinie
- ▷ `[predefined_angle]` $0 \dots 7$, entsprechend $0^\circ, 45^\circ, \dots, 315^\circ$ absolut

21.3.2 Weitere Angaben

`coeff` ist ein Faktor, mit dem die Länge der Bindung skaliert wird.



```
\chemfig{-[0]-[1]-[7,0.5,,,blue,line width=2pt]}
```

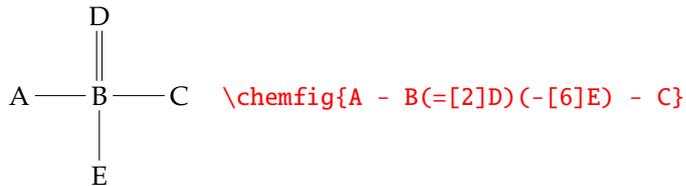
ndep, narr bezeichnet die Nummer des Atom von dem ein Bindungssymbol ausgeht, bzw. bei dem es ankommt, wenn Gruppen von Atomen ohne Bindungssymbole unter sich miteinander verbunden werden.

tikz_code wird an tikz weitergereicht, wenn die Bindung gezeichnet wird. Man kann Befehle verwenden wie z. B. Farbe (**blue**), Strichlierung (**dash pattern=on 2pt off 2pt**), Linienstärke (**line width=2pt**), unsichtbar **draw=none**, etc. Mehrere Eigenschaften durch , separieren.

Parameter, die nicht gebraucht werden, können weggelassen werden, die trennenden , müssen aber geschrieben werden.

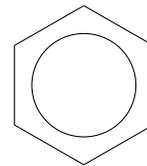
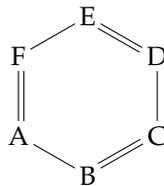
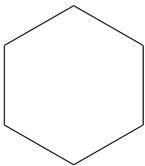
Verzweigungen

Verzweigungen werden am "Atom", von dem verzweigt wird, mit (...) angehängt, auch mehrfach und verschachtelt.



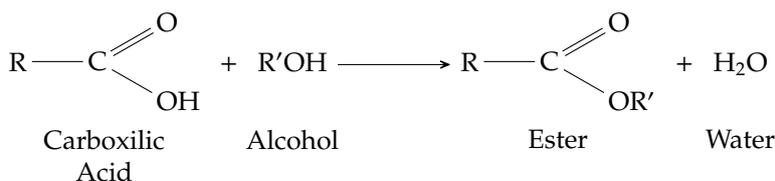
Bei komplexen Molekülen die längste Kette als Grundstruktur nehmen.

21.3.3 Ringe



`\chemfig{*6(-----)}` `\chemfig{A*6(-B=C-D=E-F=6)}` `\chemfig{**6(-----)}`

Komplexes Beispiel – Reaktionsgleichung mit Namen der Komponenten



```

\chemestart
\chemname{\chemfig{R-C(-[:30]OH)=[:30]O}}{Carboxilic\\Acid}
\+
\chemname{\chemfig{R'OH}}{Alcohol}
\arrow(.mid east--.mid west)
\chemname{\chemfig{R-C(-[:30]OR')=[:30]O}}{Ester}
\+
\chemname{\chemfig{H_2O}}{Water}
\chemestop

```

22. Farben in L^AT_EX

Farben und ihre numerische Beschreibung sind ein komplexes Thema, zum vertieften Einstieg siehe z. B. https://en.wikipedia.org/wiki/Color_space.

22.1 Farbpakete

Es gibt zwei Pakete, die Farbe zur Verfügung stellen, `color` und das etwas neuere und leistungsfähigere `xcolor`. Die wichtigsten Parameter (für `xcolor`) sind:

- ▷ **driver** kann `dvips`, `pdftex` etc. sein. Da T_EX selbst keine Farben kennt, kann nur das Tool, das im Hintergrund das Postscript- bzw. PDF-File erzeugt, die Farben verarbeiten. In der Regel braucht man den Treiber nicht angeben, wird automatisch richtig erkannt.
- ▷ **colormodel** Angabe des Farbmodells (RGB etc., s. u.), das in der Ausgabe verwendet wird. Wenn nicht angegeben, werden die Farben in dem Modell ausgegeben, in dem sie definiert wurden.
- ▷ **colornames** Mögliche Werte: `dvipsnames`, `svgnames`, `x11names`. Stellt vordefinierte Farben zur Verfügung, die über ihren Namen ansprechbar sind. Achtung, hat nichts mit `driver` zu tun, obwohl z. T. ähnlich benannt. Siehe Paketdokumentation sowie folgende URLs:
 - 68 `dvips`-Namen: <http://calque.pagesperso-orange.fr/latex/latexps.html>
 - 151 X11-Namen: http://www.sciencetronics.com/greenphotons/wp-content/uploads/2016/10/xcolor_names.pdf
 - 317 `svg`-Namen: <http://www.december.com/html/spec/colorsvg.html>
- ▷ **table** lädt `colortbl`, für farbige Zellen in Tabellen (siehe Kapitel 14.3.5, Seite 41)
- ▷ **hyperref** verbesserte Definitionsmöglichkeiten für Farben in Hyperlinks

Oft wird `xcolor` schon im Hintergrund durch ein anderes Paket geladen, das Farben braucht, und zwar mit einer bestimmten Farbnamen- oder anderen Option. Wenn man dann noch einmal mit einer anderen Option lädt, gibt es eine Fehlermeldung. In dem Fall entweder Paket nicht extra laden, oder das extra Laden vor den anderen Paketen platzieren.

22.2 Farbmodelle

In L^AT_EX sind mit `xcolor` folgende Farbmodelle verfügbar (mit TikZ und damit auch mit dem `beamer`-Paket sind nicht alle kompatibel):

rgb RGB `red` – `green` – `blue`, additive Farbmischung (Bildschirm)

cm(y)k `cyan` – `magenta` – `yellow` – (black), subtraktive Farbmischung (Druck)

hsb HSB hue (Farbton) – saturation (Farbsättigung) – brightness (Helligkeit)

HTML – RGB-Modell in hexadezimaler Schreibweise (wie in `html`)

gray Gray Grauwerte

wave Angabe von Farben für monochromatisches Licht über die Wellenlänge

`xcolor` stellt auch Befehle zur Umwandlung zwischen Farbmodellen zur Verfügung.

22.3 Farben definieren

Die Grundfarben stehen immer per Namen zur Verfügung:

- red, green, blue
- cyan, magenta, yellow
- black, gray, white, darkgray, lightgray
- brown, lime, olive, orange, pink, purple, teal, violet

Darüber hinaus die je nach Farbnamenmodell definierten. Außerdem kann man beliebig Farben selbst definieren:

`(\xglobal)\definecolor{name}{model}{colour specification}` stellt eine Farbe in Farbmodell `model` unter den Namen `name` zur Verfügung, die im dritten Parameter definiert wird. `\xglobal` stellt die Farben global zur Verfügung (bei Definition in Präambel automatisch global). Für die Spezifikation gibt es je nach `model` folgende Optionen:

- gray, rgb, hsb, cmy(k): ein, drei oder vier Werte zwischen 0.0 (schwarz bzw. keine Farbe) und 1.0 (weiß bzw. volle Farbe)
- RGB: drei ganzzahlige Werte 0 bis 255
- HSB: drei ganzzahlige Werte 0 bis 240
- HTML: drei ganzzahlige Hex-Werte 00 bis FF
- Gray: ein ganzzahliger Wert 0 bis 15
- wave: ein ganzzahliger Wert von 363 bis 814 (nm)
- named: Ein vordefinierter Farbname oder eine Kombination davon. `-name` erzeugt die Komplementärfarbe. `name1!wert!name2!wert2...` erzeugt eine Mischung der angegebenen Farben, wobei die Werte als Gewichtungsfaktoren verwendet werden. Wenn Summe der Werte <100, wird noch weiß im auf 100 fehlenden Ausmaß dazugemischt. Wenn zuletzt eine Farbe ohne Wert angegeben wird, so wird diese Farbe anstatt weiß genommen.

`\definecolorset` bzw. `\providicolorset` (siehe Paketdokumentation) stellt eine ganze Anzahl von Farben zur Verfügung.

22.4 Verwendung von Farben

22.4.1 Text in Farbe

Schalter: `\color[model]{specification}`

mit Argument: `\textcolor[model]{specification}{text}`

Die `specification` kann entweder eine volle Farbspezifikation wie oben sein, oder ein bereits definierter Name.

Beispiele:

```
\textcolor[gray]{0.8}{\rule{1cm}{2mm} A}  A
\textcolor[blue]{\rule{1cm}{2mm} A}  A
\textcolor[rgb]{0.4,0.6,0.}{\rule{1cm}{2mm} A}  A
\textcolor[hsb]{0.3,1,1}{\rule{1cm}{2mm} A}  A
\textcolor[hsb]{0.3,1,.4}{\rule{1cm}{2mm} A}  A
\textcolor[hsb]{0.3,.4,1.}{\rule{1cm}{2mm} A}  A
\textcolor{yellow!30!magenta!40}{\rule{1cm}{2mm} A}  A
\textcolor[wave]{400}{\rule{1cm}{2mm} A}  A
```

22.4.2 Box oder Seite in Farbe

`\colorbox[model]{Hintergrundfarbe}{text}`

`\fcolorbox[model]{Rahmenfarbe}{Hintergrundfarbe}{text}`

`\text` `\fboxrule2pt` `\fcolorbox{gray}{gray!20}{\color{blue}text}`

`\pagecolor[model]{specification}` Hintergrundfarbe für die ganze Seite

`\nopagecolor` um die Hintergrundfarbe aufzuheben.

23. Erzeugen und Verarbeiten von PDFs

23.1 Versionen des PDF-Standards

Es gibt verschiedene Versionen des PDF-Standards. Derzeit ist die neueste Version 1.7 bzw. seit August 2017 2.0 (siehe https://www.adobe.com/devnet/pdf/pdf_reference_archive.html für alle Details der Versionen 1.x bzw. <https://www.pdfa.org/what-will-pdf-2-0-bring/>.) Versionen ≥ 1.5 produzieren kleinere Dateien als frühere (auch 1.4 kleinere als 1.3, 1.2). Es gibt zwei wichtige PDF-Standards unabhängig von den Versionen:

- ▶ **PDF/X** – Druckvorstufe. Besondere Farbangaben und anderes, siehe <https://de.wikipedia.org/wiki/PDF/X>.
- ▶ **PDF/A** – Archivstandard, siehe <https://de.wikipedia.org/wiki/PDF/A>. Deshalb verlangt die UB, dass Abschlussarbeiten im Format A-1 (möglichst A-1b) abgeliefert werden: <http://www.boku.ac.at/bib/services/abschlussarbeiten-online/>. Die wichtigsten Merkmale von PDF/A-1 sind:
 1. Keine javascripts, nicht druckbaren Medieninhalte etc.
 2. Alle Schriften eingebettet (das gilt natürlich auch für PDF/X)
 3. Keine Elemente neuer als PDF1.4
 4. Einbettung von Metadaten wie Autor usw. in speziellen XML Feldern ist möglich und von den Bibliotheken erwünscht
 5. Einbettung von Farbmanagement-Information (ICC-Farbprofil)
 (Inzwischen gibt es neben diesem Standard PDF/A-1 noch /A-2 und A-3. A-3 ist eigentlich nicht archivtauglich. Besser bei A-1 bleiben.)
 Punkte 1 und 2 sind für L^AT_EX kein Problem.

Option 1: Spezielle `\pdf` Makros in der Präambel

Für Punkt 3 muss bei Verwendung von `pdfLATEX` unbedingt folgender Befehl in der Präambel verwendet werden: `\pdfminorversion=4`. Ohne diese Angabe wird PDF Version 1.5 erzeugt.

Metadaten kann man wie folgt angeben (es müssen nicht alle Felder vorhanden sein, und ein Teil wird automatisch von `pdfLATEX` mit vorbelegten Daten geschrieben):

```
\pdfinfo {
/Title example.pdf
/Creator TeX
/Producer pdfTeX 1.40.11
/Author Tom and Jerry
/CreationDate D:20061226154343+01'00'
/ModDate D:20061226155343+01'00'
/Subject Example
/Keywords mouse, cat }
```

(Diese Informationen können auch nachträglich verändert werden, siehe <https://wiki.univie.ac.at/display/theomet/PDF+handling>. Achtung, PDF-Metadaten sind potentielles Leak vertraulicher Information.)

Punkt 5 ist etwas mühsamer zu erfüllen, wird aber nicht strikt von der UB verlangt.

Option 2: Paket pdfx

`\usepackage[a-1b]{pdfx}` option a-1b is for PDF/A, version 1b

Und **oberhalb** von `\documentclass[options]{style}` folgendes einsetzen (weitere Metadatenkategorien nach Bedarf, siehe Paketdokumentation):

```
\begin{filecontents*}{\jobname.xmpdata}
\Title{This is the title}
\Author{First Authorname\sep Second Authorname}
\Keywords{keyword1\sep \keyword2\sep keyword3}
\end{filecontents*}
```

Mit der `filecontents*`-Umgebung wird der Inhalt der Umgebung in ein File `*.xmpdata` geschrieben, wobei `*` der Name des \TeX -Files (ohne `.tex`) ist. Daraus wird dann `pdfa-1b.xmpi` erzeugt, dessen Inhalt schließlich ins PDF kommt, und mit geeigneter Software¹ abgefragt werden kann, etwa `pdffinfo -meta myfile.pdf`. Bei Problem die `xmpdata` und `xmpi` Files löschen!

Das `pdfx`-Paket kann auch PDF/X erzeugen. In beiden Fällen braucht man ein Farbprofil-File (`.icc` oder `.icm`). Zwei Standardfiles (für /A und /X) werden mitgeliefert. Wenn sie nicht gefunden werden, unter dem verlangtem Filenamen ins Arbeitsverzeichnis kopieren².

Andere Optionen (Acrobat Professional, ghostscript)

Um PDF/A (oder /X) zu generieren, kann man auch `ghostscript` mit `-dPDFA (-DPDFX)` verwenden. Details siehe <http://ghostscript.com/doc/current/Ps2pdf.htm>.

Natürlich kann all das auch Acrobat Professional, sogar auf Knopfdruck. Damit kann man auch selbsterzeugte PDFs auf die Erfüllung der Standards prüfen. Aber Achtung, Acrobat Professional wendet bei der Erzeugung von PDFs in der Standardeinstellung für eingebettete Pixel-Grafiken ab einer bestimmten Größe verlustbehaftete Kompression und/oder Downsampling an! Auch beim Erzeugen von PDF aus PS mittels `ghostscript /ps2pdf` die Option `-dPDFSETTINGS=/printer` oder `-dPDFSETTINGS=/prepress` verwenden – bei `-dPDFSETTINGS=/screen` werden Grafiken ebenfalls verkleinert! Deshalb besser mit `pdfLATEX` erzeugte PDFs nicht von irgendjemand mit Acrobat nachbearbeiten lassen.

23.2 Einbinden (ganzer) Seiten oder mehrseitiger anderer Dokumente *als eigene Seite(n)*

Paket `pdfpages`, Verwendung:

```
\includepdf[key=val, ...]{filename}
```

Wichtigste keys:

- ▷ `pages=p1-pn`.
`pn` kann auch `last` sein. Argument wird gebraucht (sonst nur erste Seite).
 Um alle Seiten zu nehmen: `pages=-`
 So fügt z. B. `\includepdf[pages=1]{inhalt-L10.pdf}` die erste Seite des genannten Files ein, nach einem impliziten `\newpage`. Die eingefügte Seite wird mitgezählt, aber nicht umpaginiert.
- ▷ `nup=nupx x nupy`.
 Mehrere importierte Seiten verkleinert auf einer Seite anordnen (“n-up”), `nupx` ist Anzahl der Spalten, `nupy` der Zeilen.

¹Zum Beispiel das Paket <http://www.xpdfreader.com/>. Linux-User Paket `poppler-utils`.

²Zum Beispiel `sRGB_IEC61966-2-1_black_scaled.icc` aus `/usr/local/texlive/2017/texmf-dist/tex/latex/pdfx/` auf `sRGBIEC1966-2.1.icm`.

- ▶ `pagecommand={command}` Für `command` z. B. `\thispagestyle{plain}`, damit Seiten (entsprechend dem laufenden Dokument, nicht dem importierten) paginiert werden
- ▶ `\includegraphics`-Parameter wie `scale=`, `bb=`, `viewport=`, `angle=` werden übernommen!
- ▶ Weitere Parameter siehe Paketdokumentation!

Beispiel:

```
\includepdf[page=1,scale=0.9,pagecommand=\thispagestyle{plain}]{inhalt-L10}
```

23.3 Erstellung von Broschüren mit zwei Textseiten pro Druckseite

Manchmal hat man z. B. ein Dokument in A4 und möchte dieses als ein A5-Büchlein drucken, oder man hat eine Beamer-Präsentation und möchte ein Handout mit zwei oder mehr Folien pro Seite drucken.

`acroread` kann das softwaremäßig, ist aber nicht immer vorhanden, und gelegentlich liefert es auch – etwa wegen nicht perfekter Linux-Druckertreiber – nicht das gewünschte Ergebnis. Man kann mit Hilfe von `pdfpages` natürlich das gewünschte Dokument herstellen. Es gibt auch ein Linux-Kommandozeilen-Tool (auch auf Mac), das einem diese Arbeit abnimmt, und zwar das Bundle `pdfjam`.

Daraus `pdfnup`, wenn es nur ums Anordnen mehrerer Seiten geht. Beispiel: `pdfnup --nup 1x2 --no-landscape presentation.pdf` um eine Bildschirmpräsentation umzuwandeln.

`pdfbook` dient dem verkleinerten Broschürendruck (als default, zwei Hochformatseiten nebeneinander auf einer Querformatseite angeordnet). Beispiel `pdfbook --signature N input.pdf`, wobei `N` die Signaturgröße ist, d. h. die Anzahl Seiten, die anschließend ein „Heft“ ergeben soll; wenn mein input-File z. B. 31 Seiten hat, sollte man 32 angeben – es wird dann eine Leerseite angefügt.

Unter Windows kann man `pdfbox` (siehe Kap. 15.4.3, S. 49) oder `pdfbooklet` (<https://pdfbooklet.sourceforge.io/>) verwenden.

23.4 Weitere PDF-Tools

Siehe Anhang ab Seite 94. Besonders nützlich ist `pdftk` (für Linux Paket der Distribution sonst von <https://www.pdfplabs.com/tools/pdftk-the-pdf-toolkit/>)

Mit

```
pdftk input.pdf cat pagerange output output.pdf
```

kann man Seiten extrahieren. `pagerange` kann z. B. 1, 3-5, 10, 11 lauten.

Man kann auch ein File aus mehreren zusammen setzen, etwa

```
pdftk input1.pdf input2.pdf cat output output.pdf
```

Auch komplexere Optionen sind möglich.

```
pdftk input.pdf dump_data output meta.txt
```

schreibt die in einem PDF vorhandenen Metadaten in ein Textfile `meta.txt`. Man kann sie bereinigen und wieder re-importieren.

Um Metadaten gar nicht erst in ein PDF-File zu schreiben, kann man das Paket `pdfprivacy` mit den entsprechenden Optionen (siehe Dokumentation) verwenden.

24. Export / Import (html, doc, rtf, etc.)

24.1 Office-Dokumente nach L^AT_EX exportieren

Es gibt eine „Extension“ `writer2latex` für OpenOffice / Libreoffice, ist auch z. B. ein Debian-Paket. Damit kann man Office-Dokumente (Text und Tabellenkalkulation; auch MS-Office) öffnen und dann im Menüpunkt „Export“ LaTeX auswählen. Man kann wählen, wieviel der Formatierung in L^AT_EX umgesetzt werden soll – oft ist es am besten (`ultra`)clean zu nehmen und die Formatierung in L^AT_EX neu zu machen. Es gibt auch eine Commandline-Utility dazu. Eine funktionierende java-Installation auf dem Rechner wird vorausgesetzt.

24.2 L^AT_EX in HTML-, ePUB-, Office- (doc[x], odt) Format exportieren

Der Weg führt meist über HTML, da die Officeprogramme auch HTML einlesen. Auch ePub (e-Book-Format) ist HTML in einer Hülle.

Software:

`tth` <http://hutchinson.belmont.ma.us/tth> für alle Plattformen, in C geschrieben.

`latex2html` war lange Zeit die beliebteste Konvertiersoftware, hat viele gute Features (z. B. jedes Unterkapitel als eigene Datei mit Navigation). L^AT_EX-Strukturen, die nicht umgesetzt werden können, werden in png-Dateien verwandelt, die dann eingebunden werden (z. B. Mathematik). Ebenfalls für alle Plattformen (in CTAN enthalten), braucht aber eine perl-Installation.

`tex4ht`: siehe unten

Es gibt noch diverse andere Software, die ähnliche Funktionalität verspricht, aber nach meinem Kenntnisstand nicht wirklich besser ist. Aber Augen offen halten!

Für **e-Books** empfiehlt sich die Open-Source Software `calibre` (<http://calibre-ebook.com/>, bei Linux entsprechendes Paket der Distribution), die e-Books nicht nur darstellen und verwalten, sondern auch aus HTML erzeugen kann.

Alternativ bietet sich die neue Software `tex4ebook` an. Diese verwendet den konfigurierbaren, Perl-basierten Konverter `tex4ht`. `tex`-Files müssen zuvor mit `latex` (nicht `pdflatex`) übersetzt worden sein!

`pandoc` (<https://pandoc.org/>, für alle Betriebssystem) kann viele markup-Formate ineinander übersetzen.

Bei komplexeren L^AT_EX-Files ist es oft sinnvoll, verschiedene Konverter und Einstellungen zu probieren und das beste Ergebnis zu suchen.

25. Liste von weiteren Paketen und Tools

Eine relativ willkürliche und vielleicht nicht ganz aktuelle Liste von Paketen, die in der Lehrveranstaltung nicht besprochen wurden, aber auch nützlich sein können.

Überschriften, Verzeichnisse, Indizes

alnumsec, alphanum, sectsty, anonchap Formatierung von Kapitel- etc. Überschriften (Alternativen zu `titlesec`)
multitoc, shorttoc, tocloft, tovsec2 Gestaltung von Inhaltsverzeichnissen (Alternativen zu `titletoc`)
index, makeidx, makeindex, multind, forindex Erzeugen von Indizes (Stichwortverzeichnissen)

Layout

lastpage für „Seite n von m“
footmisc Fußnotengestaltung
endnotes, endnote, endnote-hy Endnoten (im Unterschied zu Fußnoten)
footbib Literaturzitate als Fußnoten
enumitem, expdlist Gestaltung von Aufzählungen (Alternative zu `paralist`)
sectionbox Farbige Boxen um Überschriften, v. a. bei Postern nützlich
setsecnum Beliebige Überschriftennummerierung, z. B. I.1.a statt 1.1.1
lineno Zeilen nummerieren
anyfontsize, fontsize einfache Bereitstellung von beliebigen Schriftgrößen bzw. beliebiger Grundschriftgröße
circledsteps beliebigen (kurzen) Text wie z. B. Zahlen mit Kreis umgeben

Tabellen, Abbildungen, Gleitumgebungen

endfloat alle Floats am Ende sammeln
rotfloat Floats rotieren
morefloats Sehr viele Floats erlauben
diagbox Tabellenzelle diagonal unterteilen 
Excel-to-LaTeX Konvertierung für Excel97

Diverses

datetime2, isodate, uhrzeit Datum und Uhrzeit in verschiedenen Formaten
commath Abkürzungen für Ableitungen u.a. mathematische Konstrukte
lhyp diverse nützliche Kleinigkeiten
appendix Gestaltung von Anhängen
fonttable eine Tabelle aller Zeichen in einem Font erstellen
prelim2e Versionen nummerieren
FIXme, todonotes Kommentare in Manuskript einfügen, auflisten

Spezifische Drucksorten

akletter Briefkopf

currvita, europasscv Lebenslauf (allgemein, bzw. EUROPASS-Stil der EU).

bizcard Visitenkarten

uebungsblatt Gestaltung von Übungsblättern

leaflet Leaflets, or what German-speaking people call “Folder”

fancyhandout Kompaktes, “fancy” layout für Handouts

paper Manuskripte für Zeitschriften, generisch

thesis und viele andere! für Diplomarbeiten und Dissertationen

wallpaper Bild als Hintergrund auf Seiten

graphpap mm-Papier

<https://ctan.org/topic/music?lang=en> Notensatz

confabs für Konferenz-Abstracts

papertex für Newsletter, Zeitungen, ...

... und viele weitere. Es lohnt sich, auf CTAN (<https://ctan.org/>) zu suchen (in Firefox kann man sich die CTAN-Suche sogar als optionale Suchmaschine einrichten). Hat man ein relevantes Paket gefunden und die Subseite dazu aufgeschlagen, so findet man unter “Suggestions” und “See also” ähnliche Pakete, sowie die “Topics”, unter denen das Paket gelistet ist – die jeweilige Themenliste ist auch ein guter Ausgangspunkt für weitere Recherchen. Bei Paketen, die seit 10 oder mehr Jahren nicht mehr aktualisiert wurden, sollte man vorsichtig sein – es kann gut sein, dass es inzwischen neuere und bessere Pakete für die jeweilige Aufgabe gibt.

DANTE-Mitglieder bekommen jedes Quartal eine Liste der neuen Pakete in der Vereinszeitschrift DIE TECHNISCHE KOMÖDIE.

Nützliche Tools

latexdiff Tool mit dem zwei \LaTeX -Dokumente verglichen werden können. Das Ergebnis ist ein `tex`-File für ein drittes Dokument. Wenn man dieses übersetzt, so werden die Unterschiede durch Farben, Unterstreichen und Durchstreichen angezeigt (ähnlich “track change”-Modus in WYSIWYG-Textprogrammen).

git-latexdiff Falls man `tex`-Dokumente mit der Versionsierungssoftware `git` verwaltet, kann man bequem ein `latexdiff` von zwei Versionen erstellen. Es gibt auch einige Pakete, die `git` und `svn` zum Beispiel durch Ausgabe von Versionsnummern unterstützen.

26. Präsentationen und Poster mit beamer

26.1 Paket beamer

beamer ist ein umfangreiches Paket, welches L^AT_EX ziemlich stark verändert und eine eigene Dokumentklasse `beamer` bereitstellt. Die logische Strukturierung erfolgt in Umgebungen namens `frame`, die jeweils eine Bildschirmseite enthalten.

Man kann Teile einer Seite (z. B. die einzelnen *items* einer Liste) schrittweise freigeben (via aufeinander folgende, sonst idente PDF-Seiten, die automatisch erzeugt werden), es gibt spezielle Kopf- und Fußzeilen, eine Möglichkeit Handouts zu erzeugen, usw.

Es gibt zahllose *themes*, die zudem anpassbar sind). Im Handbuch und auf Webseiten wie z. B. http://www.deic.uab.es/~iblanes/beamer_gallery/) kann man die anschauen.

Im Folgenden einige Beispiele, wie Seiten aussehen können:

The Problem
Approach
Properties
Solution
Outlook

New Approach – Basic Principle

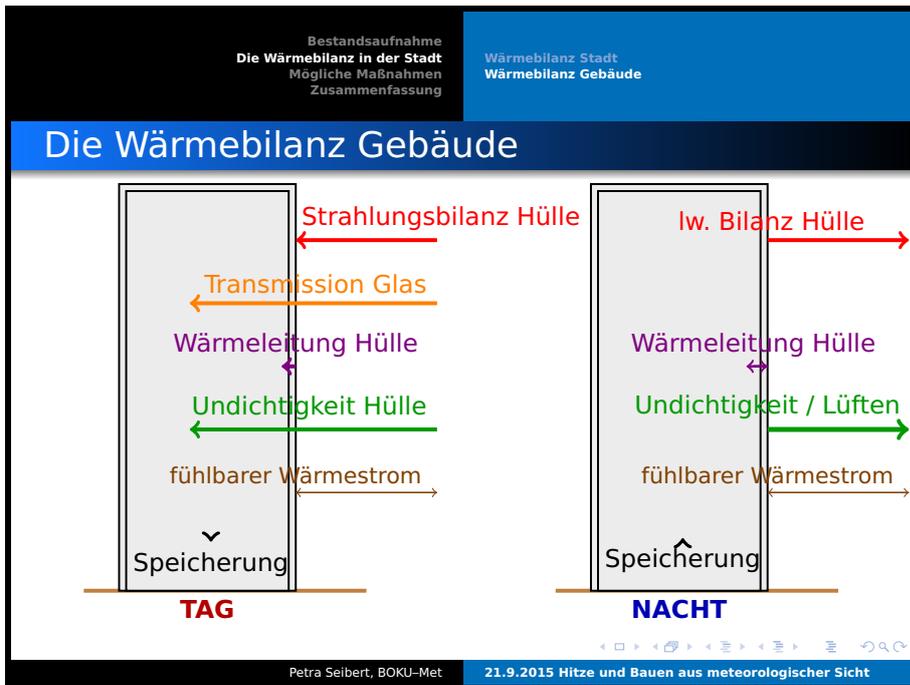
Problem can be considered as a **remapping problem**, but existing remapping algorithms, e.g. for mass-conserving semi-Lagrangian advection, do not solve this problem.

- Divide interval in three subintervals with linear interpolation
- Determine supporting point values for this subgrid
- 3 degrees of freedom:

$$\frac{1}{2} + 1 + 1 + \frac{1}{2}$$
 supporting points
 - ① integral value
 - ② central slope
 - ③ other constraints
- Requires specification of boundaries

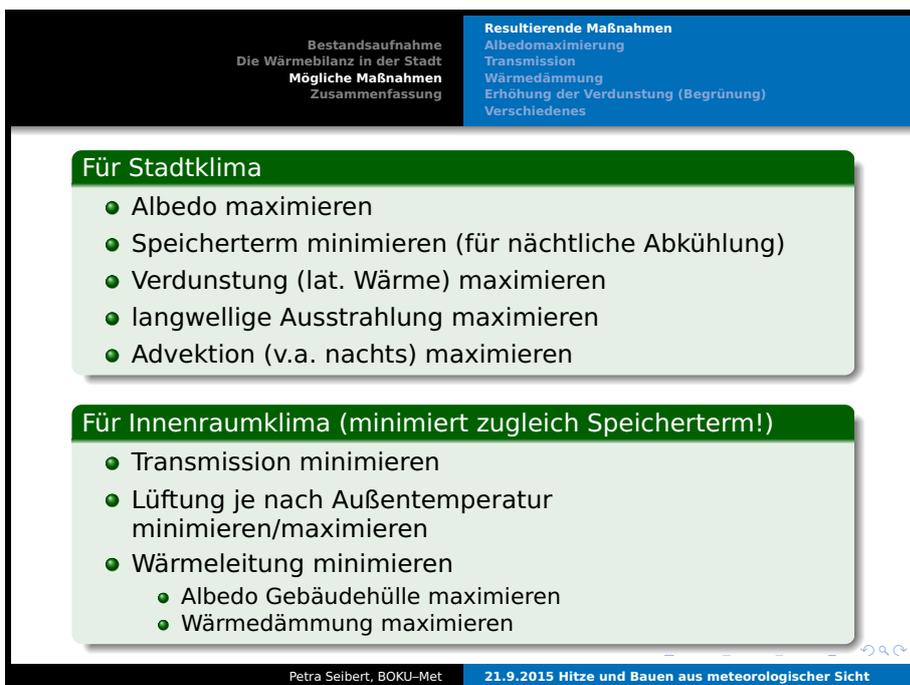
◀ ▶ ⏪ ⏩ 🔍 ↺

5
Hittmeir/Philipp/Seibert
New Interpolation Method for Gridded Extensive Variables



Hier ist die Grafik übrigens mit tikz erstellt (Auszug):

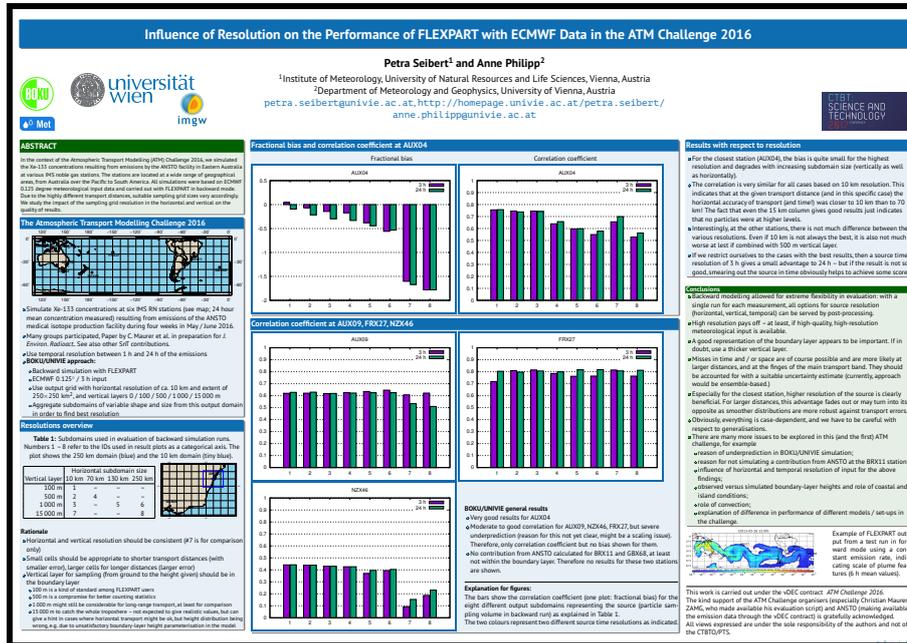
```
\usetikzlibrary{backgrounds}
\begin{tikzpicture}
\begin{scope}[xshift=190]
\draw[ultra thick,brown] (0,0) -- (4,0);\tikzstyle{arrow}=[ultra thick,->,text=]
\draw (.5,0) [thick,fill=gray!15,] rectangle (3,5.8) ;
\draw (.6,0) [thick,fill=gray!15,] rectangle (2.9,5.7) ;
\draw[arrow, color=red] (3,5) node [above] {\lw. Bilanz H\ulle} -- (5,5) ;
\draw[arrow] (1.8,.7)-- (1.8,.75) node [below] {Speicherung};
\draw[color=black!30!blue] (1.75,0) node[below] {\bfseries NACHT};
\end{scope}
\end{tikzpicture}}
```



Der Code ist hier eine normale `itemize`-Umgebung, eingeschachtelt in die `frame`-Umgebung und darin noch zwei `sampleblock`-Umgebungen (grüner Hintergrund.)

26.2 beamerposter

L^AT_EX ist für die Postererstellung sehr gut geeignet. Dazu verwendet man die Dokumentklasse **beamer** mit dem Zusatzpaket **beamerposter**. Hier ein Beispiel:



Struktur

```
\documentclass[final,size=a0]{beamer}
\usetheme{Berlin}
\setbeamercolor{palette primary}{fg=white,bg=univie} % changed this
\setbeamercolor{palette tertiary}{fg=white,bg=univie} % changed this
\setbeamercolor{block title}{bg=univie}
\setbeamerfont{block title}{fg=univie}
\setbeamerfont{block title}{series=\bfseries\sffamily}
\useinnertheme[shadow]{rounded}
\usepackage[orientation=landscape,size=a0wide,scale=1.,debug]{beamerposter}
\setbeamertemplate{caption}[numbered]
\usefonttheme{sansserif}

\title[...]{...}
%
\author[...]{...}
\institute[...]{...}

\begin{document}
\begin{frame}[plain]{}
\titlerpage

\begin{columns}[t]

\begin{column}{.25\linewidth}
\begin{exampleblock}{\large \strut \bh\bfseries ABSTRACT}
...
\end{exampleblock}
\begin{block}{\large \strut The Atmospheric Transport Modelling Challenge 2016}
...
\end{block}
\end{column}
\end{columns}
```

```

\begin{column}{.48\linewidth}
  \begin{block}{\large \strut\strut Fractional bias and correlation coefficient at AUX04}
    ...
  \end{block}
  \begin{block}{\large \strut Correlation coefficient at AUX09, FRX27, NZX46}
    ...
  \end{block}
\end{column}

\begin{column}{0.25\linewidth}
  \begin{block}{\large \strut Results with respect to resolution}
  \end{block}
  \begin{exampleblock}{Conclusions}
    ...
  \end{exampleblock}\bigskip
\end{column}

\end{columns}

\end{frame}
\end{document}

```

Zusammengefasst:

- Poster ist ein einziger frame
- Nach dem Title kommt eine columns-Umgebung
- Diese wird nach Bedarf in einzelne column-Umgebungen aufgeteilt.
- In diesen kann man die drei Blocktypen von Beamer
 - block
 - **alertblock**
 - **sampleblock**
 verwenden, oder aber auch Material ohne Block einfügen und andere L^AT_EX-Komponenten wie Listen (werden mit graphischen Aufzählungssymbolen formatiert), Boxes, Grafiken usw. verwenden.
- Die gleichmäßige Auffüllung der Spalten muss man selbst sicherstellen. Oft hilft es, die Spaltenbreite etwas zu variieren.

Vorteile:

- Spalten haben gleiche bzw. definierte Breite, Inhalte perfekt ausgerichtet
- L^AT_EX-Elemente wie Tabellen, Formeln, Literaturzitate mit BibTeX können verwendet werden
- Auch alle anderen Vorteile, wie Einbinden externer Tabellenfiles, viele Grafiken auf einmal mittels Skript einbinden, usw.
- Vektor-Grafikformate können verwendet werden, Filegröße u. U. sehr kompakt (gilt auch für Präsentationen)

HAPPY TEXING

A. PDF-Handling aus dem IMG-Wiki

Da das IMG-Wiki nicht mehr öffentlich zugänglich ist, hier als Anhang der Inhalt zum PDF-Handling (fast leere Seiten lassen sich leider nicht verhindern.)



universität
wien

PDF handling

Department of Meteorology & Geophysics

Exportiert am 27. Mai 2021

wiki.univie.ac.at



Inhaltsverzeichnis

1 Adobe Acroread pdftk ps/pdf to eps / pdf with bounding box pdfjam PDFBox pdftinfo PDF/A and PDF/X	3
2 Adobe Acroread.....	4
3 pdftk.....	5
4 ps/pdf to eps / pdf with bounding box.....	6
5 pdfjam.....	7
6 PDFBox	8
7 pdftinfo	9
8 PDF/A and PDF/X.....	10

1

- Adobe Acroread
- pdftk
- ps/pdf to eps / pdf with bounding box
- pdftjam
- PDFBox
- pdftinfo
- PDF/A and PDF/X

2 Adobe Acroread

If the GUI does not react, or not as you want, click into the open page!

3 pdftk

"The Swiss Army knife for PDF handling". Just two examples

Extract pages 4-5 from file A.pdf:

```
pdftk A.pdf cat 4-5 output B.pdf
```

Join A.pdf and B.pdf into C.pdf:

```
pdftk A.pdf B.pdf output C.pdf
```

Graphical interface to pdftk: **pdfchain**

Clean up metadata in a PDF file: first extract them:

```
pdftk A.pdf dump_data output meta
```

then edit file **meta** according to your wishes, then

```
pdftk A.pdf update_info meta output A_updated.pdf
```

4 ps/pdf to eps / pdf with bounding box

Task: make a tight bounding box (this is very useful for including the PDF into a LaTeX document) and keep/make the file small (well compressed). Solution:

epspdf -b file.ps file.pdf (-b makes the BB; operates on pdf files as well)

If you need a ps file, you can add:

pdf2ps file.pdf file.eps

pdfcrop file.pdf <outputfile.pdf>

If no outputfile is given, the output is written to `file-crop.pdf`. It is possible to use `file.pdf` also as output, then it is cropped in place. There are some useful parameters, especially **-margins N**, where N is the margin to be added in units of bp (ca. 0.37 mm).

5 pdfjam

comes with TeX Live, see <http://www.ctan.org/pkg/pdfjam>. Documentation. Most useful: pdfbook, pdfnup

Example for rearranging a presentation with 2 slides (landscape) on 1 portrait page: **pdfnup --nup 1x2 --no-landscape presentation.pdf**

To make a booklet: **pdfbook --signature N input.pdf**

6 PDFBox

The Apache PDFBox™ library is an open source Java tool for working with PDF documents. This project allows creation of new PDF documents, manipulation of existing documents and the ability to extract content from documents. Apache PDFBox also includes several command line utilities. [more](#)

7 pdftinfo

Without option, gives some simple information. With **-meta**, it writes out all metadata in xml format, including metadata provided for PDF/A compliance.

8 PDF/A and PDF/X

To produce PDF/A or PDF/X files with LaTeX, use the **pdfx** package. See package documentation for details. Note specifically that you need to provide a xmpi file either directly or by putting some code at the very top of your source file to specify the metadata. It may be necessary to either copy/link the color profile files to your working directory, or to set the path to these files

```
\pdfxSetRGBcolorProfileDir{(path to RGB color profiles)} or  
\pdfxSetCMYKcolorProfileDir{(path to CMYK profiles)} (use locate to find the  
location of the *.icc files).
```