

# **LaTeX für Penible – Mikrotypografie**

(Mit etwas Grammatik und geologischen Beispielen)

Gregor Barth

Version: 1.52 vom 24. Februar 2019

## **Inhaltsverzeichnis**

<b>1</b>	<b>Worthervorhebungen</b>	<b>4</b>
<b>2</b>	<b>Überschriften</b>	<b>5</b>
<b>3</b>	<b>Satzzeichen</b>	<b>5</b>
3.1	Abkürzungen . . . . .	6
3.1.1	Spationierung in deutschen Texten . . . . .	6
3.1.2	Allgemeiner Gebrauch von Abkürzungen . . . . .	6
3.1.3	Trennung von Abkürzungen . . . . .	7
3.1.4	Wo kein Punkt auf eine Abkürzung folgt . . . . .	7
3.1.5	Abkürzungspunkte in englischen Texten . . . . .	8
3.2	Auslassungspunkte . . . . .	9
3.3	Sonderzeichen . . . . .	10
3.3.1	Et-Zeichen (»Kaufmanns-Und«) . . . . .	10
3.3.2	Prozent und Promille . . . . .	11
3.3.3	Paragraf . . . . .	11
3.3.4	Warenzeichen und Copyright . . . . .	12
3.3.5	Genealogische Symbole . . . . .	13
3.4	Eszett . . . . .	13
<b>4</b>	<b>Striche</b>	<b>14</b>
4.1	Divis = Bindestrich . . . . .	14

4.2	Halbgeviertstrich = Gedankenstrich . . . . .	15
4.3	Geviertstrich . . . . .	16
<b>5</b>	<b>Anführungszeichen</b>	<b>17</b>
5.1	Schriftart . . . . .	17
5.2	Sprache . . . . .	17
5.3	Eingabemöglichkeiten . . . . .	18
5.3.1	Variante 1 . . . . .	18
5.3.2	Variante 2: csquotes . . . . .	20
5.3.3	Variante 3: direkte Glyphe . . . . .	21
<b>6</b>	<b>Worttrennungen und Wortabstände</b>	<b>22</b>
6.1	Manuelle Angabe der Trennstelle . . . . .	22
6.2	Trennstellen dokumentweit vorgeben . . . . .	23
6.3	Automatische Worttrennungen unterbinden . . . . .	24
6.4	Worttrennung für Fremdsprachen umschalten . . . . .	24
6.5	Manuelle Trennhilfen im BibTeX/BibLaTeX-Literaturverzeichnis . . . . .	25
6.6	Ligaturen . . . . .	26
6.7	Kursivkorrektur (Italic-Korrektur) . . . . .	27
<b>7</b>	<b>Zahlzeichen und Datumsangaben</b>	<b>29</b>
7.1	Zahlzeichen und deren Gliederung . . . . .	29
7.2	Telefon- und Faxnummern . . . . .	31
7.3	Postleitzahlen, Postfachnummern . . . . .	32
7.4	Kontonummern und Bankleitzahlen . . . . .	32
7.5	Währungsangaben . . . . .	33
7.6	Datumsangaben . . . . .	34
7.7	Uhrzeiten . . . . .	35
7.8	Minuskel- und Versalziffern . . . . .	35
<b>8</b>	<b>Mathematische Zeichen</b>	<b>36</b>
8.1	Grundrechenarten im Fließtext . . . . .	36
8.2	Hoch- und Tiefstellungen . . . . .	37
8.3	Bruchstrich, Schrägstrich . . . . .	37

8.4	Kleiner/Größer . . . . .	38
8.5	Kommata in Formeln . . . . .	38
8.6	Text in Formeln . . . . .	39
8.7	Aufrechte griechische Buchstaben . . . . .	39
<b>9</b>	<b>Nützliche Pakete für Naturwissenschaftler</b>	<b>41</b>
9.1	SI-Einheiten mit dem Paket <code>units</code> . . . . .	41
9.2	Temperaturgrad, Winkelgrad . . . . .	44
9.3	Geografische Koordinaten: Grad, Minute, Sekunde . . . . .	45
9.4	Isotopen-Schreibweise . . . . .	47
9.5	Chemische Formeln . . . . .	48
9.6	Strukturformeln . . . . .	50
9.7	$\TeX$ für Mineralogen: Miller-Indizes . . . . .	51
9.8	Ordnungszahlwörter . . . . .	51
9.9	Akademische Grade . . . . .	52

## Vorwort

Diese Zusammenfassung beinhaltet keine Wiedergabe grammatikalischer Regeln zur allgemeinen Kommasetzung, der Getrennschreibung oder Groß- und Kleinschreibung, sondern gibt anhand von Beispielen wieder, wie mikrotypografische Problemstellen im  $\TeX$ -Code angegangen werden können, um das bestmögliche Satzbild zu ermöglichen. Wo sich die Regeln auf Fremdsprachen beziehen, wird darauf hingewiesen.

### Schriftwechsel zwischen Normal-Modus und Mathe-Modus

Man beachte, dass in vielen Beispielen in den sog. Mathe-Modus gewechselt wird, dessen Schriftart ggf. nicht zur Grundschriftart im Normal-Modus passt. Bei der  $\TeX$ -Standard-Schriftart, der Computer Modern (oder in ihrer Weiterentwicklung: der Latin Modern), wird für den Mathe-Modus automatisch eine passende, in der Strichstärke und dem Duktus abgestimmte Schrift ausgewählt.

Für die Libertine als Grundschriftart gibt es, meiner Kenntnis nach, noch keine Libertine-Glyphen nur für den mathematischen Satz. Man kann aber eine passende,

sehr ähnliche Schrift für den Mathe-Modus durch folgendes Paket dazuschalten (als eines der letzten in der Präambel!):

---

```
1 \usepackage[libertine]{newtxmath}
```

---

Das funktioniert allerdings nur mit pdf $\TeX$ , soweit von mir getestet. Innerhalb von X $\TeX$  und Lua $\TeX$  bleibt dieses Paket ohne Wirkung, sodass die Formel-Schriftart substituiert wird.

Wer als Grundschriftart eine serifenlose Schrift verwendet, kann folgendes Paket laden, um serifenlose Schrift im Mathe-Modus zu erzeugen:

---

```
1 \usepackage{helvet} % Helvetica als Standard für serifenlose Schrift
2   \renewcommand{\familydefault}{\sfdefault} % aktiviert die
   ↪ Helvetica als Grundschrift für das gesamte Dokument
3 \usepackage{sfmath} % hier das Paket für serifenlose Schrift im
   ↪ Formel-Modus
```

---

## 1 Worthervorhebungen

Um Wörter hervorzuheben, verwendet man den Befehl `\emph{Wort}`. `emph` steht für *emphasize* = hervorheben, betonen. Damit wird das eingeschlossene Wort kursiviert, und innerhalb eines Absatzes aus kursiver Schrift wieder geradegestellt. Nach Tschichold werden keine Wörter kursiviert, die man besonders betonen möchte, sondern nur Eigennamen. Stattdessen bediene man sich der Anführungszeichen.

Die Worthervorhebung durch Unterstreichung ist veraltet (und wenn sie durch die Unterlängen der Buchstaben geht, sogar ästhetisch unschön); sie stammt aus Schreibmaschinen-Zeiten, als noch keine Mischung aus kursiven und geraden Buchstaben bewerkstelligt werden konnte. Obwohl Unterstreichungen (natürlich) mit  $\TeX$  hergestellt werden können, sind sie zu vermeiden.

Ebenso veraltet ist es, hervorzuhebende Wörter zu *sperren*<sup>1</sup> Richtig ist, dass Kleinbuchstaben niemals gesperrt werden, sondern allenfalls Versalien und Kapitälchen (und dann ist es fast ausnahmslos nötig!). Sperrungen können z. B. mit dem soul-Paket erreicht werden.

## 2 Überschriften

Überschriften beginnen stets mit einem Großbuchstaben, in englischen als auch in deutschen Texten.

## 3 Satzzeichen

In deutschen Texten gilt:

- Es steht immer **ein Leerzeichen nach, aber keines vor** . , ; : ? ! und schließenden Klammern. Das Leerzeichen fällt aus, wenn ein schließendes Anführungszeichen folgt.
- Es steht immer **ein Leerzeichen vor, aber keines nach** öffnenden Anführungszeichen und öffnenden Klammern.
- Es steht **nie ein Leerzeichen vor und nach** dem Schrägstrich.
- Es steht **immer ein Leerzeichen vor und nach** dem Ampersand & und dem Gleichheitszeichen =

Der Umgang mit Auslassungspunkten und Strichen (Bindestrich, Gedankenstrich, Geviertstrich) ist kontextabhängig und wird in den Abschnitten 3.2 und 4 beschrieben.

---

<sup>1</sup>Das wurde v. a. gerne bei der Frakturschrift gemacht, obwohl in diesem Fall die Auszeichnung durch Schriftmischung sinnvoller war.

## 3.1 Abkürzungen

### 3.1.1 Spationierung in deutschen Texten

Zwischen den Buchstaben der Abkürzungen setzt man (in deutschen Texten<sup>2</sup>) weder ein ganzes noch gar kein Leerzeichen, sondern ein *halbes* Leerzeichen:

- 
- 1 z.B. (FALSCH)
  - 2 z. B. (OK)
  - 3 z.\,B. (typografisch RICHTIG)
- 

ergibt:
z.B. (FALSCH)
z. B. (OK)
z. B. (typografisch RICHTIG)

### 3.1.2 Allgemeiner Gebrauch von Abkürzungen

Aus Gründen der Lesbarkeit werden Abkürzungen niemals am Satzanfang geschrieben. Z. B. beginnt dieser Satz mit einer Abkürzung. Abkürzungen, die im vollen Wortlaut gesprochen werden sowie fremdsprachige Abkürzungen erfordern einen Punkt:

- 
- 1 z.\,B.
  - 2 usw.
  - 3 Prof.
- 

Sollte die Abkürzung am Satzende stehen, wird der Punkt weggelassen (törichtes Beispiel, das durch sinnvolle Satzstellung vermieden werden kann):

---

<sup>2</sup>In englischen Texten wird weder ein ganzes, noch ein halbes Leerzeichen gesetzt, hier steht die Abkürzung *ohne* zusätzlichen Zwischenraum, beispielsweise e.g.

- 
- 1 Wir warten nur auf den Dr.. (FALSCH)
  - 2 Wir warten nur auf den Dr. (RICHTIG)
- 

Punkte bei Abkürzungen sind nicht erforderlich, sofern jeder Buchstabe einzeln gesprochen wird: EDX, ICP-MS. Schreibt man im Zusammenhang mit Maßbezeichnungen, werden Abkürzungen nur in Verbindung mit Ziffern gebraucht, ansonsten ausgeschrieben:

- 
- 1 Der Aufschluss ist 20~Meter breit. (FALSCH)
  - 2 Der Aufschluss ist 20~m breit. (RICHTIG)
  - 3 Der Bohrkern ist einige m lang. (FALSCH)
  - 4 Der Bohrkern ist einige Meter lang. (RICHTIG)
- 

### 3.1.3 Trennung von Abkürzungen

Man vergesse nicht, dass Zahl und dazugehörige Einheit niemals durch Zeilenumbruch getrennt werden sollten. Um beide beisammenzuhalten, verwende man ein *geschütztes Leerzeichen* durch eine Tilde (Beispiel-Zeile 1). Gleiches gilt für Akademische Titel (Beispiel-Zeile 2). Sowohl das halbe Leerzeichen als auch die Tilde verhindern dabei einen Zeilenumbruch an dieser Stelle. Soll dagegen ein Zeilenumbruch trotzdem ermöglicht werden, verwende man den Rückstrich mit einem folgenden Leerzeichen (Beispiel-Zeile 3).

- 
- 1 65~Ma
  - 2 Dr.~Heinrich
  - 3 Dipl.\ Geol.
- 

### 3.1.4 Wo kein Punkt auf eine Abkürzung folgt

Auf Abkürzungen von Maßen, Gewichten, Himmelsrichtungen, den meisten Währungseinheiten und chemischen Elementen sowie Formelzeichen aus den Naturwis-

senschaften folgt kein Punkt. Bei Abkürzungen von Zahlwörtern wird allerdings ein Punkt gesetzt (im Beispiel gleich mal mit den korrekten Paketen für die Abstände (siehe auch Abschnitte 9.1 und 9.5):

- 
- 1 1 g NaCl besteht aus Mrd. Atomen. (FALSCH)
  - 2 `\unit[1]{g} \ce{NaCl}` besteht aus Mrd. Atomen. (RICHTIG)
- 

### 3.1.5 Abkürzungspunkte in englischen Texten

Der Wortzwischenraum beträgt generell in allen Texten genau ein Leerzeichen. Diese Regel gilt nicht für den Zwischenraum zwischen Sätzen; sie ist von der verwendeten Sprache abhängig: Während in deutschen Texten dem Satzpunkt ebenfalls ein Leerzeichen folgt, ist der Leerraum nach dem Satzpunkt z. B. in englischen Texten größer. Mit dem Laden eines deutschen Sprachpakets (z. B. `german` oder `ngerman`) wird automatisch `frenchspacing` geladen, wodurch der Raum nach dem Satzpunkt ebenfalls ein Leerzeichen beträgt.

Nun gilt es  $\TeX$  dabei zu helfen, den Punkt einer Abkürzung nicht als Satzpunkt zu interpretieren – und damit einen unnötig großen Zwischenraum nach einem Abkürzungspunkt (in englischen Texten!) zu erzeugen.

Grundsätzlich gilt für  $\TeX$ :

- Folgt ein Punkt nach einem Großbuchstaben, ist es eine **Abkürzung**.
- Folgt ein Punkt nach einem Kleinbuchstaben, ist es der **Satzpunkt**.

Das stimmt natürlich nicht immer: Bei Abkürzungen im Satz folgt ein Punkt einem Kleinbuchstaben und ist *kein* Satzpunkt. Andererseits gibt es seltene Satzkonstruktionen, bei denen ein Großbuchstabe am Ende des Satzes steht.  $\TeX$  würde dies nicht als Satzende erkennen, und entsprechend gebe es in englischen Texten keinen vergrößerten Abstand zum nächsten Satz.

Zur Korrektur dienen Backslash und @-Zeichen:

- 
- 1 The gap is approx. 5 mm wide. (FALSCH) % *Punkt = Satzende*
  - 2 The gap is approx.\ 5 mm wide. (RICHTIG) % *Punkt = Abkürzung*



3 The sign of Superman is S. Next sentence. (FALSCH) % Punkt = Abkürzung

4 The sign of Superman is S\@. Next sentence. (RICHTIG) % Punkt =

↔ Satzende

ergibt (man beachte die unterschiedlich große Lücke nach den jeweiligen Punkten!):

The gap is approx. 5 mm wide. (FALSCH)

The gap is approx. 5 mm wide. (RICHTIG)

The sign of Superman is S. Next sentence. (FALSCH)

The sign of Superman is S. Next sentence. (RICHTIG)

### 3.2 Auslassungspunkte

Will man Auslassungspunkte setzen, tippt man nicht einfach drei Punkte, sondern verwendet `\dots` (Beispiel-Zeile 1 und 2). Wenn Auslassungspunkte für ein ausgelassenes Wort stehen oder eine kurze Lesepause einleiten sollen, setzt man vor und nach den Auslassungspunkten ein Leerzeichen. Wenn die Auslassungspunkte für einen weggelassenen Teil eines Wortes stehen, werden sie unmittelbar ohne Leerzeichen an das vorherige Wort angeschlossen (Beispiel-Zeile 3 und 4). Folgt auf Auslassungspunkte ein Satzzeichen, z. B. ein Komma, wird es direkt an die Auslassungspunkte angesetzt (Beispiel-Zeile 5 und 6). Dies gilt nicht für den Satzpunkt, dieser wird dann weggelassen (Beispiel-Zeile 7 und 8).

1 Nicht drei ... Punkte setzen! (FALSCH)

2 Nicht drei `\dots\` Punkte setzen! (RICHTIG)

3 Hieß es Velocirap `\dots`? (FALSCH)

4 Hieß es Velocirap`\dots`? (RICHTIG)

5 Wie ich weiß `\dots{}` , ist es aus. (FALSCH)

6 Wie ich weiß `\dots`, ist es aus. (RICHTIG)

7 Das Ende naht `\dots.` (FALSCH)

8 Das Ende naht `\dots` (RICHTIG)

ergibt:

Nicht drei ... Punkte setzen! (FALSCH)  
Nicht drei ... Punkte setzen! (RICHTIG)  
Hieß es Velocirap ...? (FALSCH)  
Hieß es Velocirap...? (RICHTIG)  
Wie ich weiß ... , ist es aus. (FALSCH)  
Wie ich weiß ..., ist es aus. (RICHTIG)  
Das Ende naht .... (FALSCH)  
Das Ende naht ... (RICHTIG)

Auch hier gilt, dass die Auslassungspunkte das vorherige Wort nicht durch einen Zeilenumbruch getrennt werden sollten. Entsprechend nutzt man die Tilde:

---

1 So was~\dots

---

Das Paket `ellipsis` sollte immer mit in die Präambel geladen werden, denn es korrigiert die Abstände (den Weißraum) der Auslassungspunkte zum vorherigen bzw. nächsten Wort.

### 3.3 Sonderzeichen

#### 3.3.1 Et-Zeichen (»Kaufmanns-Und«)

Das Et-Zeichen, im englischen auch »Ampersand« genannt, wird in  $\text{T}_{\text{E}}\text{X}$  mit `\&` erzeugt. Man beachte, dass es *nur* bei Firmennamen und zwischen Personen-Namen verwendet wird und nicht an jeder beliebigen Stelle, schon gar nicht als Ersatz für das Wort »und«!

- 
- 1 Hin `\&` wieder (FALSCH)  
2 Firma Schiefer `\&` Platte Inc. (RICHTIG)  
3 Hinz `\&` Meier `\&` Müller (2008) (FALSCH)  
4 Hinz `\textit{et al.}` (2008) (RICHTIG)
-

### 3.3.2 Prozent und Promille

Das Prozent-Zeichen kann über die Tastatur eingegeben werden; bei  $\LaTeX$ -Code ist allerdings darauf zu achten, dass es sich um ein Steuerzeichen handelt (das ein Kommentar einleitet) und deshalb mit einem vorangestellten Rückstrich geschrieben werden muss: `\%` ergibt `%`. Das Promille-Zeichen wird mit dem Befehl `\textperthousand` erzeugt<sup>3</sup> (Beispiel-Zeile 1).

Vor dem Promille- wie auch vor dem Prozent-Zeichen wird (in deutschen Texten<sup>4</sup>) ein *halbes Leerzeichen* eingefügt (Beispiel-Zeile 2 und 3). Bei Zusammensetzungen entfällt der Zwischenraum jedoch (Beispiel-Zeile 4 und 5), ebenso in englischsprachigen Texten:

- 
- 1 Über 9`\,\textperthousand{}` Salzgehalt.
  - 2 30`\%` beträgt der Quarzgehalt. (FALSCH)
  - 3 30`\,\%` beträgt der Quarzgehalt. (RICHTIG)
  - 4 Es gibt eine 70`\,\%ige` Zunahme. (FALSCH)
  - 5 Es gibt eine 70`\%ige` Zunahme. (RICHTIG)
- 

ergibt:

Über 9 % Salzgehalt.  
30% beträgt der Quarzgehalt. (FALSCH)  
30 % beträgt der Quarzgehalt. (RICHTIG)  
Es gibt eine 70 %ige Zunahme. (FALSCH)  
Es gibt eine 70%ige Zunahme. (RICHTIG)

### 3.3.3 Paragraf

Das Paragraf-Zeichen wird mit `\S` erzeugt und nur in Verbindung mit einer nachfolgenden Ziffer geschrieben, ansonsten wird das Wort ausgeschrieben (Beispiel-Zeile

---

<sup>3</sup>Unter Verwendung von  $\XeTeX$  und einem Unicode-konformen Zeichensatz lässt sich das Promille-Zeichen auch direkt in den Quellcode eingeben.

<sup>4</sup>In englischen Texten wird *kein* Zwischenraum zwischen Zahl und Prozent-Symbol gesetzt.

1 und 2). Da auch hier das Paragraf-Zeichen nicht von der Ziffer durch einen Zeilenumbruch getrennt werden darf, setzt man sicherheitshalber immer gleich die Tilde als *geschütztes Leerzeichen* mit (Beispiel-Zeile 3). Vor und nach dem Paragraf-Zeichen steht ein Leerzeichen. Um den Paragraf im Plural zu schreiben, erzeugt man zwei Paragraf-Zeichen hintereinander (Beispiel-Zeile 4 und 5).

- 
- 1 Der `\S` ist lang. (FALSCH)
  - 2 Der Paragraf ist lang. (RICHTIG)
  - 3 Laut `\S~6` gilt die Regel `\dots`
  - 4 Die Paragrafen 1--4 besagen `\dots` (FALSCH)
  - 5 Die `\S\S~1--4` besagen `\dots` (RICHTIG)
- 

ergibt:

Der § ist lang. (FALSCH)  
Der Paragraf ist lang. (RICHTIG)  
Laut § 6 gilt die Regel ...  
Die Paragrafen 1-4 besagen ... (FALSCH)  
Die §§ 1-4 besagen ... (RICHTIG)

### 3.3.4 Warenzeichen und Copyright

Symbole für Warenzeichen (*trademark*) und eingetragenes Warenzeichen (*registered trademark*) werden ohne Zwischenraum direkt an das Wort gesetzt (Beispiel-Zeile 1 und 2). Das *Copyright*-Zeichen wird durch einen Wortzwischenraum vom nachfolgenden Text getrennt (Beispiel-Zeile 3 und 4). Das Kommando `texttrademark` benötigt das `textcomp`-Paket. Unter Verwendung von  $\XeTeX$  und einem Unicode-konformen Zeichensatz lassen sich die Glyphen auch direkt im Quellcode eingeben.

- 
- 1 Superzement `\texttrademark\` (FALSCH)
  - 2 Superzement `\texttrademark\` (RICHTIG)

- 3 `\copyright2010 Mr.~Zement` (FALSCH)  
4 `\copyright{} 2010 Mr.~Zement` (RICHTIG)

ergibt:

Superzement™ (FALSCH)  
Superzement™ (RICHTIG)  
©2010 Mr. Zement (FALSCH)  
© 2010 Mr. Zement (RICHTIG)

### 3.3.5 Genealogische Symbole

Bei Datumsangaben stellt man häufig bestimmte Symbole voran, um ein Geburtsdatum oder Sterbedatum auszuweisen; das sind üblicherweise ein kleiner Stern und das »Toten-Kreuz« (Beispiel-Zeile 1 und 2). Beide Kommandos entstammen dem `textcomp`-Paket und werden vom Datum mit einem halben Leerzeichen spationiert.

Viele gut ausgestattete Schriftarten haben außerdem eigens dafür designte Glyphen, die ggf. erst über Unicode-konforme Module wie `XYTeX` oder `LuaTeX` zugänglich werden. Abstände innerhalb des Datum werden, wie in Abschnitt 7.6 beschrieben, erzeugt:

- 1 `\textasteriskcentered\,23.\,04. 1965 %(geboren)`  
2 `\textdagger\,11.\,03. 2004 %(gestorben)`

ergibt:

\* 23. 04. 1965  
† 11. 03. 2004

### 3.4 Eszett

Zu dem Thema, ob die Ligatur »Eszett« (ß) nur als Kleinbuchstabe existiert, bin ich der Meinung, dass ebenso das »große Eszett« (das sog. »Versaleszett«) eingesetzt werden

sollte. Dieses ist nur bei den wenigsten Schriftarten enthalten<sup>5</sup> und kann ggf. erst über Module wie Xe<sub>l</sub>TeX oder Lua<sub>T</sub>EX zugänglich gemacht werden, da diese den gesamten Glyphen-Umfang einer Schriftart ausreizen. Sollte in der verwendeten Schriftart kein Versaleszett enthalten sein, wird notfalls Doppel-S gesetzt (Beispiel-Zeile 1). Ein kleines Eszett unter Großbuchstaben ist falsch und nichts anderes (Beispiel-Zeile 2). Ein Doppel-S anstelle eines Eszett kann aber, v. a. bei Personennamen, ebenfalls falsch sein:

- 
- 1 GROSSER Text (NOTLÖSUNG)
  - 2 GROßER Text (GANZ FALSCH)
- 

Kritiker dieser verkomplizierten Anwendung befürworten die generelle Abschaffung des Eszett und seine Ersetzung durch ss. Ohnehin wird das Eszett ausnahmslos in deutscher Sprache angewendet. Allerdings ist das Eszett nicht einfach nur ein austauschbarer Buchstabe (so wie man *x* mit *ks* ersetzen könnte), sondern in seiner Hauptfunktion derjenige Indikator, der anzeigt, wie die benachbarten Buchstaben beim Sprechen zu betonen sind. Daher ist das Eszett in deutschen Texten nach wie vor unentbehrlich.

## 4 Striche

### 4.1 Divis = Bindestrich

Bindestriche werden in deutschen Texten für Bindungen und Trennungen verwendet. Der Bindestrich (Divis) ist nicht mit dem mathematischen Minus zu verwechseln! Zusammengesetzte Wörter werden gewöhnlich ohne Bindestrich geschrieben. Im Ausnahmefall wird ein Bindestrich zwischen langen, unübersichtlichen (oder nicht klar zuzuordnenden) Wörtern oder zwischen Wörtern in Verbindung mit Abkürzungen gesetzt. Muss ein Wort am Seiten-Ende getrennt werden, fügt  $\LaTeX$  automatisch den richtigen Bindestrich ein, sofern das nicht durch den Nutzer absichtlich unterbunden wird.

---

<sup>5</sup>v. a. weil es erst kürzlich in Unicode aufgenommen wurde

## 4.2 Halbgeviertstrich = Gedankenstrich

Im Gegensatz zum Bindestrich wird der *Gedankenstrich* mit - - erzeugt. Wird der Gedankenstrich zum Einschub von Text verwendet, steht er vor und nach diesem Satzfragment, jeweils mit einem Leerzeichen zwischen Gedankenstrich und Wort (Beispiel-Zeile 2); am Satzende entfällt er. Um einen Zeilenumbruch nach dem öffnenden und vor dem schließenden Gedankenstrich zu vermeiden, kann man von der Tilde gebraucht machen, sodass ein geschütztes Leerzeichen entsteht (Beispiel-Zeile 3):

- 
- 1 Basalt - ein Vulkanit - ist schwarz. (FALSCH)
  - 2 Basalt - - ein Vulkanit - - ist schwarz. (RICHTIG)
  - 3 Basalt -~-ein Vulkanit~-- ist schwarz. (RICHTIG)
- 

ergibt:

- Basalt - ein Vulkanit - ist schwarz. (FALSCH)  
Basalt – ein Vulkanit – ist schwarz. (RICHTIG)  
Basalt – ein Vulkanit – ist schwarz. (RICHTIG)

Neben dem Texteingang hat der Gedankenstrich noch andere Funktionen, und zwar als Trenner für gegenüberstehende Begriffe sowie als *Von-Bis-Strich*. Beim *Gegen-Strich* steht wie beim Texteingang davor und dahinter ein Leerzeichen (Beispiel-Zeilen 1 bis 3), beim *Von-Bis-Strich* nicht (Beispiel-Zeilen 4 bis 6). In Verbindung mit dem Wort *von*, muss das Zeichen für *bis* ausgeschrieben werden (Beispiel-Zeilen 7 bis 9). Auch um die wörtliche Rede mehrerer Personen voneinander zu unterscheiden, wird ein Halbgeviertstrich gebraucht (Beispiel-Zeilen 10 und 11):

- 
- 1 Spielstand 2-0 (FALSCH)
  - 2 Spielstand 2 - 0 (immernoch FALSCH)
  - 3 Spielstand 2 - - 0 (RICHTIG)
  - 4 10 - 14 Uhr (FALSCH)
  - 5 10-14 Uhr (FALSCH)

- 6 10--14 Uhr (RICHTIG)
  - 7 von 10--14 Uhr (FALSCH)
  - 8 von 10 bis 14 Uhr (RICHTIG)
  - 9 Probe 1--8
  - 10 »Hörst du mich?« - »Nein!« (FALSCH)
  - 11 »Hörst du mich?« -- »Nein!« (RICHTIG)
- 

ergibt:

Spielstand 2-0 (FALSCH)  
Spielstand 2 - 0 (immernoch FALSCH)  
Spielstand 2 – 0 (RICHTIG)  
10 - 14 Uhr (FALSCH)  
10-14 Uhr (FALSCH)  
10–14 Uhr (RICHTIG)  
von 10–14 Uhr (FALSCH)  
von 10 bis 14 Uhr (RICHTIG)  
Probe 1–8  
»Hörst du mich?« - »Nein!« (FALSCH)  
»Hörst du mich?« – »Nein!« (RICHTIG)

### 4.3 Geviertstrich

In deutschen Texten unüblich, in englisch- und spanischsprachigen Texten in Funktion des Gedankenstrichs (Abschnitt 4.2). Erzeugt wird der Geviertstrich mit

---

1 ---

---

Man kann den Geviertstrich außerdem zur Kennzeichnung vollkommen leerer Tabellenzellen benutzen.



## 5 Anführungszeichen

Anführungszeichen verwendet man bei der Wiedergabe von wörtlicher Rede, Zitaten, gegebenenfalls auch für Wort-Neuschöpfungen, besondere Wörter oder Namen.

Es gibt sowohl verschiedene Anführungszeichen (deutsche Gänsefüßchen; Guillemets, nach innen oder außen gedreht ... – je nach Land sehr unterschiedlich!) als auch verschiedene Möglichkeiten, diese einzugeben. Das hängt von der verwendeten Schriftart (Fraktur/Antiqua), Sprache (deutsch, englisch, französisch) als auch dem Compiler ( $\LaTeX$ ,  $X_{\LaTeX}$ ) ab.

### 5.1 Schriftart

Bei Verwendung einer Fraktur-Schriftart (z. B. mit dem `yfonts`-Paket) werden ausschließlich Gänsefüßchen verwendet; die Chevrons gehören zur Antiqua.

### 5.2 Sprache

In deutschen (Antiqua-)Texten sind sowohl die „Gänsefüßchen“ als auch die sog. »Chevrons« (mit den Spitzen zum Wort zeigend) erlaubt; letztere sollten vorgezogen werden, da sie sich besser in den Fließtext einfügen und keine optische Lücke reißen. Man beachte, dass die Gänsefüßchen nicht einfach zwei Striche sind, sondern nur in der Form 99–66 als solche gültig sind, d. h. die anführenden sehen aus wie zwei normale Kommata, die abschließenden Anführungszeichen sind dasselbe, nur um 180 Grad gedreht. (Die englischen doppelten Anführungszeichen haben die Form: 99 (gespiegelt)–99, und beide stehen oben!)

Tschichold ist der Meinung, dass auch einfache  $\langle$  Chevrons  $\rangle$ , nach außen weisend, zur Kennzeichnung wörtlicher Rede völlig ausreichend sind. Ich meine allerdings, dass sie für diese Aufgabe nicht auffällig genug sind, und sich allenfalls zur Hervorhebung einzelner Wörter im Fließtext eignen. Warum aber dann nicht kombinieren? – Doppelte Chevrons für die wörtliche Rede und einfache Chevrons für einzelne, hervorzuhebende Wörter? Verwendet man einfache Chevrons, dann tut ihnen ein zwischen Wort und Anführungszeichen gestelltes halbes Leerzeichen sehr gut:  $\langle \backslash, \text{Wort} \backslash, \rangle$  Das ist allerdings abhängig von der Schriftart und der mitgelieferten Zurichtung: Bei der *Libertine* halte ich ein halbes Leerzeichen für angebracht, bei der *Junicode* ist das meiner

Meinung nach nicht notwendig.

Von einfachen ‚Gänsefüßchen‘ sollte man Abstand nehmen, da zumindest das eingehende Anführungszeichen mit einem Komma verwechselt werden kann. Die deutschen Anführungszeichen werden immer direkt und ohne Leerzeichen an das jeweilige Wort oder den jeweiligen Satz angeschlossen.

Englische Anführungszeichen entsprechen einem einfach oder doppelt hochgestellten “Gänsefüßchen” (amerikanische vs. britische Schreibweise), wobei die einfache Variante bevorzugt werden sollte – sie reißt keine so große optische Lücke in den Text.

### 5.3 Eingabemöglichkeiten

#### 5.3.1 Variante 1

Deutsche Gänsefüßchen (doppelt) mittels `\glqq` (*german-left-qq*) bzw. `\grqq` (*german-right-qq*) (Beispiel-Zeile 1), oder für die Chevrons (doppelt) mittels `\flqq` (*french ...*) und `\frqq` (diese müssen dann vertauscht werden, um nach innen zu weisen, d. h. *french-right* vorne und *french-left* hinten!, Beispiel-Zeile 2):

- 
- 1 `\glqq` Rede mit mir!`\grqq`
  - 2 `\frqq` Rede mit mir!`\flqq`
- 

ergibt:

„Rede mit mir!“  
»Rede mit mir!«

Folgt dieser Eingabeweise noch Text, muss ein Rückstrich oder leeres Klammerpaar nachgestellt werden, sonst klebt das Folgewort am abführenden Anführungszeichen:

- 
- 1 Das soll ein `\glqq` Beispiel`\grqq{}` sein?
-

ergibt:

Das soll ein „Beispiel“ sein?

Wenn nach beispielsweise wörtlicher Rede sowieso ein Satzzeichen folgen soll, lässt man das Befehlszeichen oder das leere Klammerpaar natürlich weg:

- 
- 1 `\glqq` Das ist fossil!`\grqq`, sagte er.
  - 2 Am Ende folgt ein `\glqq` Beispiel`\grqq`.
- 

ergibt:

„Das ist fossil!“, sagte er.  
Am Ende folgt ein „Beispiel“.

Wer innerhalb des Bereichs mit Anführungszeichen nochmal etwas *anführen* möchte, verwendet halbe Anführungszeichen und lässt beim Kommando einfach ein q weg:

- 
- 1 `\glqq` Das ist fossil!`\grqq`, sagte er.
  - 2 `\glqq` Ein `\glq` Granit`\grq\` ist das!`\grqq`
- 

ergibt:

„Das ist fossil!“, sagte er.  
„Ein ‚Granit‘ ist das!“

Sollten die normalen Anführungszeichen und die halben Anführungszeichen einmal direkt nebeneinander stehen, werden diese mit einem kleinen Zwischenraum getrennt. Diese Anpassung erfolgt allerdings mit der `babel`-Option `german/ngerman` automatisch. Andernfalls setzt man manuell ein halbes Leerzeichen zwischen `\glqq` und `\glq`.

Englische Anführungszeichen werden nur wie folgt gesetzt (deutsches Tastaturlayout): Vordere Anführungszeichen mit `2 × Umschalt+Akzent` (Taste rechts neben

Fragezeichen); hintere Anführungszeichen mit 2 × Umschalt+Apostroph (Taste rechts neben Ä):

- 
- ```
1 ``A fine-grained sandstone.`` (DOPPELT)
2 `A fine-grained sandstone.` (EINFACH)
```
- 

ergibt:

“A fine-grained sandstone.” (DOPPELT)  
‘A fine-grained sandstone.’ (EINFACH)

### 5.3.2 Variante 2: csquotes

Die Nutzung des csquotes-Pakets ist vermutlich die bequemste und fehlerfreiste Variante zur Eingabe von Anführungszeichen. Hierfür wird gleichnamiges Paket in die Präambel geladen:

- 
- ```
1 \usepackage[babel,german=guillemets]{csquotes}
```
- 

wobei die Option `german=guillemets` vorgibt, dass die »deutsche« Version der französischen Anführungszeichen verwendet werden soll<sup>6</sup>. Im Fließtext wird das einzuschließende Wort dann einfach mit `\enquote{ }` umgrenzt:

- 
- ```
1 \enquote{Das ist ein schöner Tag.}, sagte Erna zu Paul.
```
- 

ergibt:

»Das ist ein schöner Tag.«, sagte Erna zu Paul.

Die Nutzung des csquotes-Pakets bringt folgende Vorteile mit sich:

---

<sup>6</sup>Die Paket-Dokumentation bietet viele weitere Optionen je nach Nationalität und den dort geläufigen Anführungszeichen, z. B. `english=british`.

- Die Anführungszeichen treten immer paarweise auf, d. h. es kann nicht, wie beim manuellen Setzen, aus Versehen ein Anführungszeichen vergessen werden. Was mit dem `\enquote{ }`-Kommando eingefasst wird, hat also tatsächlich vorne und hinten ein Anführungszeichen; wird dagegen die geschweifte Klammer vergessen, gibts beim Kompilieren eine Fehlermeldung.
- Unterordnete Anführungszeichen, also Anführung innerhalb einer Anführung, werden automatisch gesetzt und dabei herabgestuft. Innerhalb der `enquote`-Umgebung wird einfach eine weitere benutzt.
- Wie die Paketoption schon anzeigt, kann die Art der Anführungszeichen sehr leicht für das gesamte Dokument geändert werden. Beispielsweise transformiert die Option `german=quotes` statt `german=guillemets` *alle* Anführungszeichen im Dokument korrekt in deutsche Gänsefüßchen (Gänsefüßchen unten und oben).

Es gibt aber auch Nachteile:

- Wird eine `enquote`-Umgebung direkt neben einem von Text umflossenen `wrapfig`-Bild verwendet, kommt es zu Komplikationen in der Darstellung. Hier muss man die `enquote`-Umgebung wieder auflösen und das anführende und abführende Zeichen von Hand setzen, z. B. mit `\frqq` und `\flqq` oder direkter Glyphen-Eingabe (Abschnitt 5.3.3).
- Will man den Quellcode wieder in eine herkömmliche Textverarbeitung transferieren, muss man von Hand alle eingefügten `enquote`-Befehle tilgen.
- `csquotes` funktioniert momentan nicht zusammen mit dem `polyglossia`-Paket, das bei  $X_{\text{E}}\text{TeX}$  oder  $\text{LuaTeX}$  anstelle von `babel` als Sprachverwaltung verwendet wird. Allerdings lassen sich mit  $X_{\text{E}}\text{TeX}$  und  $\text{LuaTeX}$  Unicode-Glyphen, beispielsweise die Anführungszeichen, direkt im Quellcode eingeben.

### 5.3.3 Variante 3: direkte Glyphe

Wie ich an anderer Stelle empfehle, sollte die Dokument-Zeichencodierung immer auf Unicode eingestellt sein. Folgende Zeile kommt dafür in die Präambel:

---

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

---

Dadurch wird ermöglicht, dass man Anführungszeichen auch *direkt* in den Quellcode als Glyphen einfügen kann, also z. B. »Wort« oder „Beispiel“. Diese Regel gilt zunächst für Anführungszeichen; viele andere Glyphen kann man zwar auch direkt eingeben, werden dann aber beim Kompilieren anders (oder gar nicht) dargestellt als sie sollten. Was Anführungszeichen angeht, macht es jedoch keinen Unterschied, ob man » oder `\frqq` schreibt.

Man beachte, dass man die deutschen Gänsefüßchen erst aus einer Glyphenliste der Schriftart herauskopieren muss, ehe man sie in den Quelltext einfügen kann; die gewohnte Eingabe über Umschalt+2 (das Tastenkürzel, das in einer herkömmlichen Textverarbeitung wie Word zu einem deutschen Gänsefüßchen interpretiert wird) erzeugt nur ein ASCII-Anführungszeichen ("), und das ist in jedem Fall falsch.

Im Unterschied zum `csquotes`-Paket muss man außerdem selbst darauf achten, dass die Anführungszeichen immer paarweise auftreten.

Vorteilhaft an der direkten Glyphen-Eingabe ist außerdem, dass der Quellcode übersichtlicher wird, weil er nicht so viele Befehlsanweisungen enthält.

## 6 Worttrennungen und Wortabstände

### 6.1 Manuelle Angabe der Trennstelle

Normalerweise macht das  $\text{\TeX}$  mit der Worttrennung (Silbentrennung) ganz prima, vorausgesetzt, das richtige Sprachpaket wurde geladen (man kann nicht erwarten, dass ein deutscher Text richtig getrennt wird, wenn das französische Sprachpaket in die Präambel geladen wurde, siehe Abschnitt 6.4!). Allerdings gibt es in jeder Sprache einige exotische Wörter und solche aus der Fachsprache, die  $\text{\TeX}$  nicht kennt und eventuell an falscher Stelle oder gar nicht trennt. Stehen diese Wörter am Zeilenende, und die Trennung erfolgt nicht, kann es zu sog. `overflow`-Boxen kommen, da die Text-Box eines Absatzes nicht mehr optimal begrenzt wird. Für solche Fälle gibt man manuelle Trenn-Regeln vor:

Kennt  $\LaTeX$  beispielsweise das Wort `Latit` nicht, kann man die Trenn-Positionen manuell vorgeben, indem man im Quelltext schreibt:

---

```
1 La\ -tit
```

---

Man beachte, dass in diesem Fall das Wort wirklich nur an *dieser* Stelle getrennt wird bzw. an jenen Stellen, wo `\ -` steht.

Falls die Trennung nur zum Teil korrekt ist, kann man ebenfalls nachhelfen:

---

```
1 Myrm-ek-tit (FALSCHE TRENNUNG nach LaTeX)
2 Myr\ -mek-tit (KORREKTE TRENNUNG)
```

---

Aber jetzt ist zu beachten, dass  $\LaTeX$  die internen Trennregeln für das Wort deaktiviert und *nur* an der vorgegebenen Stelle trennt, demnach `Myr-mektit` (was ja ebenfalls falsch ist)! Aus diesem Grund gebe man entweder *alle* Stellen an, an denen getrennt werden soll (demnach `Myr\ -mek\ -tit`) oder man verwende den Befehl für optionales Trennen, nämlich `"` -

In diesem Fall schreibt man:

---

```
1 Myr" -mektit
```

---

Nun behält  $\LaTeX$  seinen internen Trennmechanismus bei, weiß aber, dass es im ersten Wortteil, falls nötig, an genau dieser Stelle zu trennen hat.

$\LaTeX$  kennt noch viele weitere dieser Kommandos, mit denen man beispielsweise einen Umbruch setzen kann, ohne dass ein Bindestrich erzeugt wird. Eine Übersicht gibt [diese](#) Seite.

## 6.2 Trennstellen dokumentweit vorgeben

Tritt ein Wort, das problematisch zu trennen ist, sehr oft im Dokument auf, kann man seine Trennung global festlegen:

---

```
1 \usepackage{hyphenat}
2 ...
3 \hyphenation{Ba-salt Gra-nit An-de-sit} % (einzelne Wörter durch
   ↪ Leerzeichen trennen!)
```

---

Damit wird erreicht, dass im ganzen Dokument die Wörter Basalt, Granit und Andesit immer an den gegebenen Stellen getrennt werden (Achtung! *Alle* Trennstellen müssen vorgegeben werden!).

### 6.3 Automatische Worttrennungen unterbinden

Soll vollständig verhindert werden, dass ein Wort oder eine Wortgruppe (zum Beispiel Telefonnummern) durch Trennregeln am Zeilenende »zerrissen« werden, kann man den `\mbox{}`-Befehl verwenden:

---

```
1 \mbox{(0455) 2304-5543}
```

---

Auch bei Verweisen auf Bilder und Tabellen sind geschützte Leerzeichen angebracht, die mit einer Tilde erzeugt werden. So werden Fig./Tab. und ihre Ziffer am Zeilenende nicht getrennt.

---

```
1 Wie in Tab.~\ref{tab01intro} zu sehen ist, stimmen die Daten mit
   ↪ Diagramm in Fig.~\ref{fig05data} überein.
```

---

### 6.4 Worttrennung für Fremdsprachen umschalten

Wird an einem mehrsprachigen Dokument gearbeitet, sollten die Worttrennungen nicht nur von einem Sprach-Paket durchgeführt werden. Das kennt jeder, der einen deutschen Text mit englischem Sprachpaket kompiliert, oder umgekehrt. – Die Worttrennungen sind dann falsch. Es hat also Sinn, einzelnen Absätzen die verwendete



Sprache mitzuteilen, sodass TeX die korrekten Trennungsalgorithmen anwenden kann.

Im einfachsten Fall wurde das Dokument größtenteils in einer Sprache (z. B. Deutsch) verfasst, und enthält fremdsprachige Einschaltungen (z. B. Englisch). In diesem Fall müssen beide Sprachen geladen werden:

---

```
1 \usepackage[UKenglish,ngerman]{babel}
```

---

Die Reihenfolge der geladenen Sprachen ist dabei *nicht willkürlich*! Es wird die Hauptsprache des Dokuments (ngerman) als *letzte* (hier: zweite) Option geladen!

Anschließend können fremdsprachige Einschaltungen gekennzeichnet werden. Innerhalb dieser Umgebung wird der Trennungsalgorithmus der angegebenen Fremdsprache benutzt:

---

```
1 Dies ist mein deutscher Haupt-Text.  
2  
3 \begin{otherlanguage}{UKenglish}  
4 In this paragraph I write English. The English hyphenation patterns  
  ↪ are used.  
5 \end{otherlanguage}
```

---

Das babel-Paket kennt noch viele weitere Kommandos, beispielsweise zur Kennzeichnung einzelner Wörter, die als Fremdsprache erkannt werden sollen. Für die meisten Fälle reicht die oben beschriebene Methode jedoch aus.

## 6.5 Manuelle Trennhilfen im BibTeX/BibLaTeX-Literaturverzeichnis

Das bisher Gesagte zum Thema manuelle Trennvorgaben gilt zunächst nur für den Dokument-Quelltext, nicht aber für das Literaturverzeichnis, wenn es automatisch mithilfe von BibTeX oder BibLaTeX gesetzt wurde. Denn auch hier kann es Fachwörter geben, die sich dem Standard-Trenn-Algorithmus entziehen, infolgedessen das Wort

über den Satzblock hinaussteht und als Warnung beim Kompilieren eine *Bad Box* gemeldet wird.

Die einfachste Lösung wäre, eine manuelle Trennstelle in der .bib-Datei vorzugeben. Doch ist das eigentlich nicht im Sinne der Literaturverwaltung, zumal die .bib-Datei als Quelle für verschiedene Literaturverwaltungsprogramme genutzt werden kann.

Stattdessen bietet sich als Lösung an, die manuellen Trennstellen in der Literatur-Metadatei einzugeben, die beim Kompilieren mit BibTeX oder BibLaTeX aus der .bib-Datei erzeugt wird. Diese Metadatei hat die Endung .bbl. Darin finden sich alle im Dokument zitierten Literatureinträge, wie sie schließlich im Literaturverzeichnis gesetzt werden sollen.

Der Nachteil der Bearbeitung der .bbl-Datei ist, dass die Korrekturen *nach* dem letzten Kompilieren mit BibTeX oder BibLaTeX durchgeführt werden müssen. Denn sonst würde die .bbl neu erzeugt und alle vorgenommenen Änderungen überschrieben werden!

Sind in der .bbl-Datei alle Trennstellen vorgegeben worden, wird das Dokument durch nochmaligen Durchlauf des TeX-Kompilers gesetzt. Nun sollten auch die problematischen Wörter im Literaturverzeichnis korrekt umgebrochen werden.

## 6.6 Ligaturen

Eine Ligatur ist eine Verschmelzung eines (häufig auftretenden) Buchstabenpaars zu einer Glyphe. In deutschen Texten dürfen Ligaturen aber nicht an jeder Stelle eingesetzt, sondern müssen an Wortfugen (Morphem-Grenzen) aufgelöst werden. Weitere Informationen zu diesem Thema habe ich [hier](#) zusammengestellt.

Die Standardligaturen `f i` und `f l` sollten in jeder Antiqua-Schriftart enthalten sein, und können auch meistens bedenkenlos gesetzt werden. Zur Aktivierung dieser Standardligaturen muss man nichts weiter tun; sie werden beim Kompilieren automatisch gesetzt.

Zum Unterbinden (Unterdrücken) von Ligaturen kann man sich des Befehls `\/` bedienen, der aber eigentlich der sog. Kursivkorrektur dient (Abschnitt 6.7) und daher zuweilen übermäßige Buchstabenabstände erzeugt (zusätzliches Kerning wird eingefügt) (Beispiel-Zeile 2). Korrekt (aber auch umständlich) wäre die Verwendung von `\kern0pt`, das zwischen die Ligatur gesetzt wird (Beispiel-Zeile 3).

Für deutsche Texte (das `german-` oder `ngerman-`Paket muss geladen sein!) wird stattdessen der Befehl `"|` benutzt (Beispiel-Zeile 4). Damit wird nicht nur die Ligatur aufgelöst, sondern gleichzeitig ein Wortumbruch erlaubt, falls das Wort am Zeilenende steht.

- 
- 1 Auflage, Kaufhaus, affig (typografisch FALSCH)
  - 2 Auf\l/age, Kauf\l/haus, aff\l/ig (bedingt OK)
  - 3 Auf\kern0pt lage, Kauf\kern0pt haus, aff\kern0pt ig (KORREKT, aber  
↔ umständlich)
  - 4 Auf"|lage, Kauf"|haus, aff"|ig (typografisch RICHTIG)
- 

ergibt:

Auflage, Kaufhaus, affig (typografisch FALSCH)  
Auflage, Kaufhaus, affig (bedingt OK)  
Auflage, Kaufhaus, affig (KORREKT, aber umständlich)  
Auflage, Kaufhaus, affig (typografisch RICHTIG)

Bei starkem Zoom an die Ergebnisbox sieht man, dass es in jeder Zeile minimale Unterschiede hinsichtlich des Buchstabenabstands zwischen Ligatur-Buchstabenpaaren gibt. Man achte darauf, wie unnatürlich groß der Abstand wird, wenn `\l/` benutzt wird (2. Zeile).

Selbstverständlich bedeutet diese Nacharbeit, d. h. die manuelle Kontrolle jeder Ligatur im Dokument und ggf. deren Auflösung, einen erheblichen Aufwand. Auf meiner Webseite gebe ich [hier](#) eine Whitelist zum Abgleich von Ligaturen. Dort wird auch auf das `selnolig`-Paket verwiesen (nur bei `Lua $\TeX$` ), das bekannte Ligaturen automatisch unterdrückt.

## 6.7 Kursivkorrektur (Italic-Korrektur)

Wird eine Schrift kursiviert, erhalten nahezu alle Buchstaben eine zur Normalschrift abweichende Form. Einige von ihnen erhalten eine dermaßen verstellte Ober- oder Unterlänge (z. B. *f* vs. *f*), dass sie mit vor- oder nachstehenden Zeichen kollidieren

können. Typische Problemstellen sind Klammern und die kursiven Minuskeln *y* und *f* (siehe Beispiele). Da aus typografischer Sicht die Überschneidung zweier Zeichen inakzeptabel ist, muss eine Korrektur durch zusätzlichen Zwischenraum erfolgen; das nennt man »Kursiv-Korrektur«.

Prinzipiell entstehen keine Probleme, wenn man das Makro `\textit{Wort}` benutzt. Das korrigiert nämlich den Abstand bereits. Keine Korrektur erfolgt dagegen bei Gebrauch des Makros `\itshape`, mit dem ganze Textpassagen kursiviert werden können. Die Verwendung von `\` schafft dabei einen geringen Abstand, sodass die Zeichen nicht länger kollidieren:

- 
- 1 `[y-Achse, Schlaf]` (Normalschrift)
  - 2 `{\itshape [y-Achse, Schlaf]}` (unkorrigiert)
  - 3 `{\itshape [\y-Achse, Schlaf\]}` (manuell korrigiert)
  - 4 `\textit{[y-Achse, Schlaf]}` (automatisch korrigiert)
- 

ergibt:

`[y-Achse, Schlaf]` (Normalschrift)  
`[y-Achse, Schlaf]` (unkorrigiert)  
`[y-Achse, Schlaf]` (manuell korrigiert)  
`[y-Achse, Schlaf]` (automatisch korrigiert)

Die Beispiele der oben gezeigten Ergebnisbox mögen nicht überzeugend sein; die Abstände zwischen Buchstaben und anderen Zeichen hängen u. a. stark von der verwendeten Schriftart ab. Gut bearbeitete Schriften erfahren eine ausführliche Zurichtung (*kerning*), bei der die Zeichenpaare bereits vorsorglich einen Abstand erhalten haben; bei der hier verwendeten Libertine-Schriftart ist das der Fall.

Zusammenfassend kann gesagt werden: Werden im Dokument Text-Passagen mit `\itshape` kursiviert, sollte man diese auf Klammern prüfen.

## 7 Zahlzeichen und Datumsangaben

### 7.1 Zahlzeichen und deren Gliederung

Sätze und Überschriften mit einer Ziffer (= Zahlzeichen) zu beginnen, sollte vermieden werden (Ausnahme: Kapitel-Nummerierung). Weiterhin gilt nach einem ungeschriebenen Gesetz, dass Zahlzeichen bis 12 (oder 13) als Worte auszuschreiben seien, alles darüber in Ziffern (das gilt natürlich nicht für Datumsausdrücke, siehe dort):

- 
- 1 fünfundachtzig Tage (FALSCH)
  - 2 85 Tage (RICHTIG)
  - 3 4 Stunden (OK)
  - 4 vier Stunden (BESSER)
- 

Ziffern kleiner als 10 werden nur dann als Ziffern geschrieben, wenn

- ihnen eine Einheit vorausgeht oder folgt: 5 kg Gestein
- sie eine bestimmte Stelle in einer nummerierten Folge darstellen: Wert Nummer 8
- sie im Vergleich mit Zahlen größer 10 im selben Satz stehen<sup>7</sup>: 4 von 22 Proben Gestein

Die Ziffern Null und Eins werden trotzdem als Worte gesetzt, wenn sie dadurch leichter zu verstehen sind:

- 
- 1 Der Wert mit Null ist Nord.
- 

Ganze Zahlen, die aus mehr als vier Ziffern bestehen, werden von der Endziffer (Einer) aus in dreistellige Gruppen zerlegt. Dazwischen steht ein halbes Leerzeichen. Gleiches gilt für Nachkommastellen, wobei vom Komma aus in Dreiergruppen gegliedert wird. Tausendertrenner wie Punkte sind möglich (aber erst ab 10 000 sinnvoll), sollten aber

---

<sup>7</sup>Eigentlich sollten alle Zahlzeichen, die in einem Vergleich verwendet werden, als Ziffern geschrieben werden!

mit einem halben Leerzeichen gesetzt sein. Als Dezimaltrenner fungiert in deutschen Texten ausschließlich das Komma, in englischen Texten der Punkt!

- 
- 1 14352657 (OK)
  - 2 14\,352\,657 (BESSER)
  - 3 1,4352657 (OK)
  - 4 1,435\,265\,7 (BESSER)

---

| ergibt:              |
|----------------------|
| 14352657 (OK)        |
| 14 352 657 (BESSER)  |
| 1,4352657 (OK)       |
| 1,435 265 7 (BESSER) |

Handelt es sich um Zahlenfolgen zwischen Null und Eins, wird *eine* Null vor das Komma gestellt:

- 
- 1 00,235 (FALSCH)
  - 2 ,235 (FALSCH)
  - 3 0,235 (RICHTIG)

---

| ergibt:         |
|-----------------|
| 00,235 (FALSCH) |
| ,235 (FALSCH)   |
| 0,235 (RICHTIG) |

Wie immer gilt: Zwischen einer Ziffer und einer nachfolgenden Einheit darf kein Zeilenumbruch erfolgen, d. h. man setzt ein geschütztes (halbes) Leerzeichen ein oder verwendet gleich das `unit`-Paket (siehe Abschnitt 9.1):

- 
- 1 3\,kg Gestein
-

Wenn einer Zahl ein einzelner Buchstabe folgt, der Teil einer Nummerierung ist, wird zwischen diesen ein halbes Leerzeichen gesetzt<sup>8</sup>:

- 
- 1 In Abb.~5b sieht man `\dots` (OK)
  - 2 In Abb.~5 b sieht man `\dots` (OK)
  - 3 In Abb.~5\,b sieht man `\dots` (BESSER)
- 

ergibt:

- In Abb. 5b sieht man ... (OK)  
In Abb. 5 b sieht man ... (OK)  
In Abb. 5 b sieht man ... (BESSER)

Steht die Kombination Abb.~5 am Zeilenende, verhindert die Tilde ggf. einen sinnvollen Zeilenumbruch, d. h. mit Tilde kann eine mehr oder weniger auffällige *Bad Box* erzeugt werden. In diesem Fall beuge man sich den Regeln guter Typografie und behandle die Zusammengehörigkeit zwischen Abb. und Abbildungsziffer nicht so streng. Mit anderen Worten, man lasse dann die Tilde weg, sodass ordentlich getrennt werden kann.

## 7.2 Telefon- und Faxnummern

Telefon- und Faxnummern werden von der letzten Ziffer ausgehend in Zweiergruppen mit einem halben Leerzeichen gegliedert. Davon unabhängig wird auch die Ortskennzahl in Zweiergruppen von der letzten Ziffer ausgehend gegliedert. Sie kann in runde Klammern eingefasst werden (zwischen ihr und der Durchwahl steht dann ein Leerzeichen) oder wird mit einem Schrägstrich oder Bindestrich von der Durchwahl abgetrennt:

- 
- 1 0456123456789 (OK)
  - 2 0456 123456789 (BESSER)

---

<sup>8</sup>Abbildungsreferenzierungen dieser Art sollte man allerdings eigentlich vom internen  $\LaTeX$ -Automatismus erledigen lassen.

- 3 0\,45\,61\,23\,45\,67\,89 (noch BESSER)
  - 4 04\,56/1\,23\,45\,67\,89 (GUT)
  - 5 (04\,56)~1\,23\,45\,67\,89 (AM BESTEN)
- 

ergibt:

0456123456789 (OK)  
0456 123456789 (BESSER)  
0 45 61 23 45 67 89 (noch BESSER)  
04 56/1 23 45 67 89 (GUT)  
(04 56) 1 23 45 67 89 (AM BESTEN)

### 7.3 Postleitzahlen, Postfachnummern

Postleitzahlen werden in Deutschland nicht gegliedert. Postfachnummern werden von rechts nach links in Zweiergruppen gegliedert; dazwischen steht ein halbes Leerzeichen:

- 
- 1 Postfach 12345 (OK)
  - 2 Postfach 1 23 45 (BESSER)
  - 3 Postfach 1\,23\,45 (AM BESTEN)
- 

ergibt:

Postfach 12345 (OK)  
Postfach 1 23 45 (BESSER)  
Postfach 1 23 45 (AM BESTEN)

### 7.4 Kontonummern und Bankleitzahlen

Kontonummern werden von hinten in Dreiergruppen gruppiert, dazwischen steht ein halbes Leerzeichen:



- 
- 1 Konto 12345678 (OK)
  - 2 Konto 12 345 678 (BESSER)
  - 3 Konto 12\,345\,678 (AM BESTEN)
- 

ergibt:

Konto 12345678 (OK)  
Konto 12 345 678 (BESSER)  
Konto 12 345 678 (AM BESTEN)

Die nationale Bankleitzahl ist 8-stellig und wird von links nach rechts in zwei Dreiergruppen und eine Zweiergruppe gegliedert; zwischen den Gruppen steht ein halbes Leerzeichen:

- 
- 1 BLZ 12345678 (OK)
  - 2 BLZ 123 456 78 (BESSER)
  - 3 BLZ 123\,456\,78 (AM BESTEN)
- 

ergibt:

BLZ 12345678 (OK)  
BLZ 123 456 78 (BESSER)  
BLZ 123 456 78 (AM BESTEN)

## 7.5 Währungsangaben

Handelt es sich um die Währung Euro, wird der Euro-Betrag vom Cent-Betrag mit einem Komma getrennt. Beide Teileinheiten, ebenso Betrag und Währungszeichen dürfen durch einen Zeilenumbruch nicht getrennt werden. Bei Beträgen mit mehr als vier Stellen wird der Betrag von hinten in Dreiergruppen gegliedert, zwischen denen ein halbes Leerzeichen steht. Sofern das Euro-Zeichen in der verwendeten Schriftart enthalten ist, kann es leicht durch den Befehl `\texteuro`<sup>9</sup> oder – bei Gebrauch

<sup>9</sup>Das `textcomp`-Paket muss geladen sein.

von LuaTeX oder XeTeX! – die Eingabe der entsprechenden Unicode-Glyphe erzeugt werden:

- 
- 1 25000,43 Euro (OK)
  - 2 25 000,43 Euro (BESSER)
  - 3 25\,000,43~\texteuro\ (AM BESTEN)
- 

ergibt:

25000,43 Euro (OK)  
25 000,43 Euro (BESSER)  
25 000,43 € (AM BESTEN)

## 7.6 Datumsangaben

Im Fließtext sollte, wenn möglich, der Monat eines Datums immer ausgeschrieben werden. Schreibt man das Datum nur mittels Ziffern, setzt man zwischen Tag und Monat ein *halbes* Leerzeichen, zwischen Monat und Jahr ein *ganzes*. Eine zweistellige Jahreszahl ist nur dann erlaubt, wenn deren Interpretation eindeutig ist. Sie wird in diesem Fall mit einem *halben* Leerzeichen vom Tag/Monat-Block abgegrenzt:

- 
- 1 29.06.1981 (OK)
  - 2 29. 06. 1981 (BESSER)
  - 3 29.\,06. 1981 (formal KORREKT)
  - 4 29.\,06.\,81 (ebenfalls KORREKT)
- 

ergibt:

29.06.1981 (OK)  
29. 06. 1981 (BESSER)  
29. 06. 1981 (formal KORREKT)  
29. 06. 81 (ebenfalls KORREKT)

Von–Bis-Angaben erfolgen bei Jahreszahlen durch einen Schrägstrich:

---

1 2004/05

---

## 7.7 Uhrzeiten

Bei den Uhrzeiten wird die Anzahl der Stunden, Minuten und Sekunden mit je zwei Ziffern angegeben. Bei ungefähren Angaben schreibt man nur die Stunden:

- 
- 1 4:31:6 (FALSCH)
  - 2 4:31:06 (halb FALSCH)
  - 3 04:31:06 (KORREKT)
  - 4 Er kommt etwa 19:00 Uhr. (OK, aber nicht sinnvoll)
  - 5 Er kommt etwa 19 Uhr. (BESSER)
- 

## 7.8 Minuskel- und Versalziffern

Minuskelziffern (auch Mediävalziffern) sind im Gegensatz zu Versalziffern so geschnitten, dass sie eine Ober- und Unterlänge haben (wie der Buchstabe g oder h im Serifen-Text). Versalziffern sind dagegen immer genau gleich groß und haben die Höhe eines Versal-Buchstabens.

Im Fließtext sollten normalerweise Minuskelziffern eingesetzt werden, da sich deren Ober- und Unterlängen besser in den Gesamttext einfügen und ermüdungsfreier (da weniger ablenkend) gelesen werden können. Versalziffern sollten dagegen vorrangig in Tabellen und Daten-Matrizen zur Anwendung kommen; einige Schriftarten stellen sogar speziell geschnittene, d. h. gleich breite Tabellenziffern zur Verfügung, die erst bei Verwendung von  $X_{\text{T}}\text{E}_{\text{X}}$  oder  $\text{LuaT}_{\text{E}}\text{X}$  zugänglich werden.

Ob sein Dokument mit Minuskelziffern ausgestattet werden kann, hängt von der verwendeten Schriftart ab: Sind keine im Glyphensatz enthalten, können logischerweise auch keine dargestellt werden. Wird die  $\text{E}_{\text{T}}\text{E}_{\text{X}}$ -Standardschriftart *Computer Modern* oder *Latin Modern* benutzt, genügt das Einbinden des Pakets `hfoldsty` in die Präambel.

Wer die *Libertine*-Schriftart bevorzugt, aktiviert das über die Paket-Option:

- 
- ```
1 \usepackage[osf]{libertine} %bei pdfLaTeX
2 \setmainfont[Numbers=OldStyle]{Linux Libertine O} %bei XeTeX und
  ↪ LuaTeX
```
- 

## 8 Mathematische Zeichen

### 8.1 Grundrechenarten im Fließtext

Um kürzere Gleichungen im Fließtext unterzubringen, muss man nicht immer gleich in die Formelumgebung wechseln. Denn oft besteht der Fall, dass die im Fließtext verwendete Schriftart nicht zu der der Formelumgebung passt und ggf. durch eine andere substituiert wird (siehe Vorwort).

Als Geowissenschaftler bediene ich mich nur selten komplexen Formeln, sondern eher im Fließtext untergebrachten Formel-Abschnitten, Symbolen, Einheiten und Rechenzeichen. Die Kommandos `\texttimes`, `\textminus` und `\textdiv` stammen aus dem `textcomp`-Paket. Für komplexe Formeln sollte man auf jeden Fall in den Formel-Modus wechseln, da hier auch die Abstände zwischen den Rechenzeichen, Ziffern und Symbolen technisch perfekt gesetzt werden.

- 
- ```
1 2 x 3 - 4 / 2 = Vier. (FALSCH)
2 2 \texttimes{} 3 \textminus{} 4 \textdiv{} 2 = Vier. (BESSER)
3 2\,\texttimes\,3\,\textminus\,4\,\textdiv\,2 = Vier. (noch BESSER)
```
- 

ergibt:

$2 \times 3 - 4 / 2 = \text{Vier. (FALSCH)}$

$2 \times 3 - 4 \div 2 = \text{Vier. (BESSER)}$

$2 \times 3 - 4 \div 2 = \text{Vier. (noch BESSER)}$

## 8.2 Hoch- und Tiefstellungen

Für Hoch- oder Tiefstellungen von Ziffern (Exponenten) oder Buchstaben im *Fließtext* muss nicht zwingend in die Formelumgebung gewechselt werden. Sie können mit den beiden Kommandos `\textsubscript` (Tiefstellung) und `\textsuperscript` (Hochstellung) erreicht werden. Das Kommando `\texttimes` wird über das `textcomp`-Paket bereitgestellt. Mit XeTeX und LuaTeX können die entsprechenden Unicode-Glyphen natürlich auch direkt in den Code eingesetzt werden (Beispiel-Zeile 3):

- 
- ```
1 20 x 20 qm (FALSCH)
2 20 \texttimes\ 20\,m\textsuperscript{2} (KORREKT)
3 1,435 \texttimes{} 10\textsuperscript{17} km3 (KORREKT mit
   ↪ XeTeX/LuaTeX)
```
- 

ergibt:

20 x 20 qm (FALSCH)  
20 × 20 m<sup>2</sup> (KORREKT)  
1,435 × 10<sup>17</sup> km<sup>3</sup> (KORREKT mit XeTeX/LuaTeX)

## 8.3 Bruchstrich, Schrägstrich

Vor und nach dem schrägen Bruchstrich wird *kein* Leerzeichen gesetzt (Beispiel-Zeilen 1 und 2); dies gilt auch für Wörter, die mit dem Schrägstrich getrennt werden (Beispiel-Zeilen 3 und 4). Für mathematische Brüche im Fließtext empfiehlt sich das `\unitfrac`-Kommando (siehe Abschnitt 9.1).

- 
- ```
1 4 / 5 (FALSCH)
2 4/5 (KORREKT)
3 ja / nein (FALSCH)
4 ja/nein (KORREKT)
```
-

ergibt:

4 / 5 (FALSCH)  
4/5 (KORREKT)  
ja / nein (FALSCH)  
ja/nein (KORREKT)

#### 8.4 Kleiner/Größer

Vor und nach dem größer/kleiner-Zeichen (> und <) wird im Fließtext ein ganzes Leerzeichen gesetzt.

- 
- 1 mehr>als nichts (FALSCH)
  - 2 mehr > als nichts (RICHTIG)
  - 3 1<8 (FALSCH)
  - 4 1 < 8 (RICHTIG)
- 

ergibt:

mehr>als nichts (FALSCH)  
mehr > als nichts (RICHTIG)  
1<8 (FALSCH)  
1 < 8 (RICHTIG)

#### 8.5 Kommata in Formeln

In Formel-Umgebungen werden Kommata für gewöhnlich wie Text-Kommata verwendet, d. h. um Wörter zu trennen. Der dabei erzeugte kleine Abstand nach dem Komma ist bei Dezimalzahlen natürlich falsch (Beispiel-Zeile 1). Das Komma wird deshalb in geschweifte Klammern eingefasst (Beispiel-Zeile 2):

- 
- 1 \$7,1 + 2,6 = 9,7\$ (FALSCH)
  - 2 \$7{,}1 + 2{,}6 = 9{,}7\$ (RICHTIG)

---

ergibt:

7,1 + 2,6 = 9,7 (FALSCH)

7,1 + 2,6 = 9,7 (RICHTIG)

Wie ich gelesen (aber noch nicht ausprobiert) habe, erledigt die korrekte Darstellung des Kommas durch das Laden des `ziffer`-Pakets:

---

```
1 \usepackage{ziffer}
```

---

## 8.6 Text in Formeln

Manchmal will man innerhalb der Formelumgebung ein paar kleine Kommentare in Textform unterbringen. Nur unter Verwendung des `text`-Kommandos werden die eingegebenen Buchstaben nicht als Formelzeichen interpretiert:

---

```
1 $24{,}7~\% (Schmelzanteil)$ (FALSCH)
```

```
2 $24{,}7~\%~\text{(Schmelzanteil)}$ (RICHTIG)
```

---

ergibt:

24,7 %(Schmelzanteil) (FALSCH)

24,7 % (Schmelzanteil) (RICHTIG)

## 8.7 Aufrechte griechische Buchstaben

Normalerweise erreicht man die Integration griechischer Buchstabe durch die Eingabe in der Formelumgebung:

---

```
1 $\alpha = \Theta + \omega + \epsilon - \Gamma$
```

---

ergibt:

$$\alpha = \Theta + \omega + \epsilon - \Gamma$$

Man beachte, dass die griechischen Buchstaben ggf. kursiv (und ggf. in einer anderen Schriftart!) gesetzt werden, da sie innerhalb der Mathe-Umgebung als Variable interpretiert werden. Das ist manchmal nicht gewünscht, da sie in vielen Fällen kein Formelzeichen repräsentieren sollen (siehe Beispiele unten).

Für diese Zwecke gibt es die Pakete `upgreek` und `textgreek`, die im Grunde dasselbe tun: Aufrecht stehende, griechische Buchstaben aus der Grundschriftart zu erzeugen. Hierfür muss man aber *dennoch* in die Mathe-Umgebung wechseln, stellt dem Kommando aber ein `up` voran, das je nach Groß- (Up) oder Kleinschreibung (`up`) einen großen oder kleinen griechischen Buchstaben erzeugt:

- 
- 1 `\alpha = \beta + \gamma` (UNSCHÖN)
  - 2 `\upalpha = \upbeta + \upgamma` (BESSER)
  - 3 Sedimente des Lias `\alpha` 1. (UNSCHÖN)
  - 4 Sedimente des Lias `\upalpha` 1. (BESSER)
  - 5 Kluft-Messwerte beginnen mit `\Uptheta`. (KORREKT)
- 

ergibt:

$$\alpha = \beta + \gamma \text{ (UNSCHÖN)}$$

$$\alpha = \beta + \gamma \text{ (BESSER)}$$

Sedimente des Lias  $\alpha$  1. (UNSCHÖN)

Sedimente des Lias  $\alpha$  1. (BESSER)

Kluft-Messwerte beginnen mit  $\Theta$ . (KORREKT)

Beim 2. erwähnten Paket, `textgreek`, kann man dagegen auf die Umschaltung in die Mathe-Umgebung verzichten, v. a. wenn es um Inline-Anwendung geht. Ein zweiter Vorteil ist, dass der griechische Buchstabe die Formatierung des Absatzes annimmt. Wenn also z. B. ein ganzer Absatz mit Fett formatiert wird, wird auch der griechische Buchstabe fett:



- 
- ```

1 \usepackage[euler]{textgreek}
2 ...
3 Schichten des Lias \textalpha\ 1. (KORREKT)
4 {\textbf{Schichten des Lias \textalpha\ 1.}} (nun auch in FETT)

```
- 

ergibt:

Schichten des Lias  $\alpha$  1. (KORREKT)  
**Schichten des Lias  $\alpha$  1.** (nun auch in FETT)

Groß- und Kleinbuchstaben werden erzeugt, indem man den Anfangsbuchstaben des Buchstabenwortes groß oder kleinschreibt:

- 
- ```

1 \textdelta, \textkappa % Kleinbuchstaben
2 \textGamma, \textTheta % Großbuchstaben

```
- 

ergibt:

$\delta, \kappa$   
 $\Gamma, \Theta$

Wer z. B. auf einem GNU/Linux-System seine TeXLive-Pakete aus den Paketquellen bezieht, findet das textgreek-Paket im Paket texlive-latex-extra; zusätzlich muss das Paket texlive-lang-greek installiert sein.

Nutzer von XeTeX und LuaTeX haben es einfacher, denn sie können durch die Unicode-Kompatibilität einen griechischen Buchstaben aus dem Glyphensatz der Schriftart, sofern enthalten, direkt in den Quellcode eingeben.

## 9 Nützliche Pakete für Naturwissenschaftler

### 9.1 SI-Einheiten mit dem Paket units

Wer glaubt, die Angabe von SI-Units, also Standard-Einheiten wie Meter, Sekunde usw. sei einfach eine Aneinanderreihung von Buchstaben, ggf. im Mathematik-Modus,

der irrt: Und zwar gilt nach internationaler Konventionen, dass a) die Schriftart der Einheiten in Roman sein soll, und b) aufrecht stehend ist. Im Mathematik-Modus erfolgt ja die automatische Kursivierung der Buchstaben<sup>10</sup>:

- 
- 1 `$m^2$` (FALSCH)
  - 2 `m\textsuperscript{2}` (RICHTIG)
- 

| ergibt:         |
|-----------------|
| $m^2$ (FALSCH)  |
| $m^2$ (RICHTIG) |

Hinzu kommt, dass der Abstand zwischen Anzahl und Einheit nicht willkürlich gewählt wird, sondern einem halben Leerzeichen entspricht (entsprechen soll)<sup>11</sup>. Außerdem dürfen Anzahl und Einheit über das Zeilenende nicht getrennt werden.

Um sich nun die ganze Arbeit mit Abstandsregeln, Schrägstellung usw. zu sparen, sei das Paket `units` empfohlen, womit das Kommando `\unit` verfügbar gemacht wird. Darin wird in der eckigen Klammer (Option) der Wert angegeben, in der geschweiften Klammer (Argument) steht die Einheit (Beispiel-Zeilen 4 und 5). Quadrat oder Kubik können direkt über die Tastatur eingegeben werden. Soll die Potenz eine höhere Ziffer als drei repräsentieren, existieren je nach Umfang der verwendeten Schriftart kleine hochgestellte Zahlen, z. B. <sup>5</sup> oder <sup>9</sup>. Hoch- oder Tiefstellungen werden, wie in Abschnitt 8.1 beschrieben, mit den Kommandos `textsubscript` und `textsuperscript` erreicht. Alternativ können die im Glyphensatz enthaltenen hoch- oder tiefgestellten Ziffern direkt eingegeben werden, sofern man einen Unicode-kompatiblen Compiler wie  $\text{Xe}\text{L}\text{A}\text{T}\text{E}\text{X}$  oder  $\text{Lua}\text{L}\text{A}\text{T}\text{E}\text{X}$  verwendet.

Es wird davon abgeraten, das `\unit`-Kommando auf den Grad-Celsius-Befehl (siehe Abschnitt 9.2) anzuwenden (Beispiel-Zeile 6). Die richtigen Abstände würden in diesem Fall falsch gesetzt werden (das `units`-Paket korrigiert in der Tat nur den Abstand zwischen Zahl und Einheit, achtet aber nicht darauf, ob die Einheit Sinn ergibt!).

---

<sup>10</sup>Man verwechsle nicht Einheiten mit Formelzeichen! Einheiten sind aufrecht zu schreiben, Formelzeichen stehen kursiv!

<sup>11</sup>»Was zusammengehört, wird auch enger zueinander gesetzt.«

- 
- 1 Der Dike ist 3 m breit. (OK)
  - 2 Der Dike ist `$3 m$` breit. (FALSCH)
  - 3 Der Dike ist `3\,m` breit. (KORREKT)
  - 4 Der Dike ist `\unit[3]{m}` breit. (auch KORREKT)
  - 5 Die Fläche beträgt `\unit[44]{km2}`.
  - 6 `\unit[7]{\textcelsius}` sind viel zu kalt. (FALSCH)
  - 7 `7\textcelsius{}` sind viel zu kalt. (RICHTIG)
- 

ergibt:

Der Dike ist 3 m breit. (OK)  
Der Dike ist *3m* breit. (FALSCH)  
Der Dike ist 3 m breit. (KORREKT)  
Der Dike ist 3 m breit. (auch KORREKT)  
Die Fläche beträgt 44 km<sup>2</sup>.  
7°C sind viel zu kalt. (FALSCH)  
7°C sind viel zu kalt. (RICHTIG)

Das units-Paket kennt ein weiteres Kommando, und zwar `\unitfrac`, das wie `\unit` für den Fließtext gedacht ist. In abgesetzten Formeln wird stattdessen das `\frac{}{}`-Kommando gebraucht.

- 
- 1 Die Absink-Rate beträgt `2\,m/s`. (OK)
  - 2 Die Absink-Rate beträgt `\unitfrac[2]{m}{s}`. (BESSER)
- 

ergibt:

Die Absink-Rate beträgt 2 m/s. (OK)  
Die Absink-Rate beträgt 2 m/s. (BESSER)

**Achtung!** Das units-Paket und seine Kommandos scheinen in der Nähe einer subfig-Umgebung Probleme zu machen (Fehlermeldungen beim Kompilieren und falsche Darstellung des gesetzten Textes).

## 9.2 Temperaturgrad, Winkelgrad

Wer sich nicht anders zu helfen weiß, trickst ein Grad-Zeichen über ein hochgestelltes *circ* (Kringel) im Mathematik-Modus – eine unschöne Variante (Beispiel-Zeilen 1 und 3). In vielen modernen Schriftarten gibt es dafür eine eigene Glyphen, die typografisch korrekt gezeichnet ist: Das »Grad Celsius« ist nicht einfach nur ein Kringel und ein »C« dahinter, sondern beide bilden eine Einheit und stehen enger zueinander als Kringel und Wert. Dasselbe gilt für das einfache Grad-Zeichen (z. B. für Winkelgrad). Beide im folgenden gezeigten Kommandos, `\textcelsius` und `\textdegree`, entstammen dem `textcomp`-Paket. Ein den Befehlen nachgestellter Rückstrich bewahrt das notwendige Leerzeichen zum folgenden Wort. `\textcelsius` funktioniert nicht unter  $\text{\TeX}$ . Alternativ können in  $\text{\TeX}$  und  $\text{\LuaTeX}$  auch direkt die Glyphen in den Quellcode eingefügt werden. Ein eventuelles negatives Vorzeichen sollte im Fließtext mittels `\textminus` (ebenfalls aus dem `textcomp`-Paket) erzeugt werden:

- 
- 1 `55$^{\circ}$C` sind heiß. (UNSCHÖN)
  - 2 `55\textcelsius\` sind heiß. (KORREKT)
  - 3 `90$^{\circ}$` sind ein rechter Winkel. (UNSCHÖN)
  - 4 `90\textdegree\` sind ein rechter Winkel. (KORREKT)
  - 5 `$$ 70\textcelsius\` sind saukalt. (UNSCHÖN und FALSCH)
  - 6 `-70\textcelsius\` sind saukalt. (immer noch FALSCH)
  - 7 `\textminus 70\textcelsius\` sind saukalt. (FAST richtig)
  - 8 `\textminus70\textcelsius\` sind saukalt. (KORREKT)
-

ergibt:

55°C sind heiß. (UNSCHÖN)

55°C sind heiß. (KORREKT)

90° sind ein rechter Winkel. (UNSCHÖN)

90° sind ein rechter Winkel. (KORREKT)

– 70°C sind saukalt. (UNSCHÖN und FALSCH)

-70°C sind saukalt. (immer noch FALSCH)

– 70°C sind saukalt. (FAST richtig)

-70°C sind saukalt. (KORREKT)

### 9.3 Geografische Koordinaten: Grad, Minute, Sekunde

In den Geowissenschaften dürfte es häufiger vorkommen, dass man geografische Koordinaten in der Schreibweise: Grad – Minuten – Sekunden eingeben muss. Wie beim Winkelgrad lässt sich die Grad-Koordinate mit einem `\textdegree` versehen. Bei der Minuten- und Sekunden-Koordinate aber einfach nur ein bzw. zwei Apostroph, oder gar deutsche Gänsefüßchen zu setzen, ist typografisch falsch!

Je nach Ausbau der Schriftart existieren dafür spezielle Glyphen (Sekunden- und Minutenzeichen, und zwar auf den Unicode-Positionen U+2032 und U+2033). Man könnte sie in jedem UTF8-konformen Quellcode einfach einfügen, indem man die entsprechenden Glyphen aus einer Zeichentabelle herauskopiert. Je nachdem, ob man mit `pdfTeX`, `XTeX` oder `LuaTeX` weiterarbeitet, können die Unicode-Glyphen direkt dargestellt werden (Beispiel-Zeile 7) oder werden ggf. substituiert. Die ganze Sache ist also nicht so einfach ...

Die Beispiel-Zeilen 1 und 2 zeigen die Imitation des Grad-Zeichens mit einem hochgestellten Kringel in der Formelumgebung, Minute und Sekunde werden imitiert mit einem Apostroph bzw. Doppel-Apostroph oder ASCII-Anführungszeichen. Beide Varianten sind unbrauchbar und typografisch falsch.

Die Beispiel-Zeilen 3 bis 5 zeigen eine alternative Eingabeweise, wobei die Befehle `\textquotesingle` (Minute-Glyphe) und `\dq` (Sekunde-Glyphe) ebenfalls aus dem `textcomp`-Paket entnommen sind. In Beispiel-Zeile 4 wird die Glyphe für Minute und Sekunde im Mathe-Modus erzeugt und dann hochgestellt. Das ist akzeptabel, aber

immer noch umständlich einzugeben. In Zeile 5 werden die entsprechenden Glyphen direkt über ihre Unicode-Position angesprochen<sup>12</sup>. Das Ergebnis ist dasselbe.

Nun lässt sich diskutieren, was davon die korrekte Schreibweise ist. Aus typografischer Sicht ist die (doppelte) Hochstellung eines »Formel-Apostrophs« Unsinn (Beispiel-Zeile 4). Üblicherweise sind Minuten- und Sekundenzeichen auch nicht so stark geneigt<sup>13</sup> wie das Formel-Apostroph.

Am einfachsten arbeitet man mit dem `\ang`-Befehl aus dem `siunitx`-Paket (Beispiel-Zeile 6). Allerdings werden ohne Mitgabe von Paket-Optionen auch keine Leerzeichen zwischen den Werten gesetzt (typografisch falsch), sodass man am Dokumentanfang noch folgendes Kommando mitgibt:

---

```
1 \ssetup{arc-separator = \,} %für ein halbes Leerzeichen
```

---

```
1 10∘ 44' 36" N (UNSCHÖN und FALSCH)
2 10∘ 44' 36' ' N (UNSCHÖN und FALSCH)
3 10∘ 44' 36" N (UNSCHÖN und FALSCH)
4 10∘ 44' 36" N (AKZEPTABEL)
   ↪ 36' (AKZEPTABEL)
5 10∘ 44' 36" N (AKZEPTABEL)
6 \ang{10;44;36} N (KORREKT)
7 10∘ 44' 36" N %direkte Eingabe der Unicode-Glyphen
```

---

<sup>12</sup>Hierfür muss noch das `ucs`-Paket in die Präambel geladen werden!

<sup>13</sup>Abhängig von der verwendeten Schriftart.

ergibt:

10° 44' 36"N (UNSCHÖN und FALSCH)

10° 44' 36" N (UNSCHÖN und FALSCH)

10° 44' 36" N (UNSCHÖN und FALSCH)

10° 44' 36'' (AKZEPTABEL)

10° 44' 36'' N (AKZEPTABEL)

10° 44' 36'' N (KORREKT)

10° 44' 36'' N

#### 9.4 Isotopen-Schreibweise

Bei geowissenschaftlichen Texten kann es vorkommen, dass ab und zu Isotopen zu schreiben sind. Wie bekannt ist, wird ein Isotop dergestalt geschrieben, dass die Massenzahl links oben vor dem Element-Kürzel steht, die Protonenzahl dagegen links unten von diesem. Es spricht nichts dagegen, das Isotop mithilfe einer Formel-Umgebung darzustellen (das Element-Kürzel ist dann aber kursiviert, siehe Beispiel-Zeile 1), oder sich der `\textsubscript`- und `\textsuperscript`-Befehle zu behelfen (Beispiel-Zeile 2). Professioneller und in seiner Schreibweise kürzer ist es, ein speziell für die Isotopen-Schreibweise gedachtes `isotope`-Paket zu nutzen (Beispiel-Zeilen 4 und 5):

- 
- 1 `$_{37}^{87}\text{Rb}$` (UNSCHÖN)
  - 2 `\textsuperscript{87}\textsubscript{37}\text{Rb}` (UNSCHÖN)
  - 3
  - 4 `\usepackage{isotope}`
  - 5 `\isotope[37][87]{Rb}` (KORREKT)
-

ergibt:

${}^{87}_{37}\text{Rb}$  (UNSCHÖN)

${}^{87}_{37}\text{Rb}$  (UNSCHÖN)

${}^{37}_{87}\text{Rb}$  (KORREKT)

## 9.5 Chemische Formeln

Chemische Formeln enthalten im einfachsten Fall tiefgestellte Ziffern, die die Zahlenverhältnisse der Elemente wiedergeben, z. B.  $\text{H}_2\text{O}$ . Wie Tiefstellungen erreicht werden, steht in Abschnitt 8.2. Werden die Formeln allerdings komplexer, kommen Pfeile und Beschriftungen dazu, wird es komplizierter. Dann sei das Paket `mhchem` empfohlen, das einem alle Arbeit abnimmt. Das Paket *muss* mit folgender Option geladen werden:

---

```
1 \usepackage[version=3]{mhchem}
```

---

Hier nun einige Beispiele mit dem `\ce`-Kommando; weitere Beispiele in der Paket-Dokumentation:

---

```
1 H$_2$O (UNSCHÖN)
2 H\textsubscript{2}O (OK)
3 \ce{H2O} (BESSER)
4
5 AgCl$_2^-$ (UNSCHÖN)
6 \ce{AgCl2-} (BESSER)
7
8 (NH$_4$)$_2$S (UNSCHÖN)
9 \ce{(NH4)2S} (BESSER)
10
11 CO$_2$+C $\rightarrow$ 2CO (UNSCHÖN)
```



```

12 \ce{CO2 + C -> 2CO} (BESSER)
13
14 \ce{CO2 + C <=> 2CO} % Doppelpfeil
15 \ce{$A$ ->[\ce{+H2O}] $B$} % mit Text auf Pfeil
16 \ce{SO4^2- + Ba^2+ -> BaSO4 v} % mit Pfeil für Ausfällung
17
18 \ce{CO2 (Gas) + C -> 2CO} % mit Text in Formel, unschön
19 \ce{CO2}~\ce{\text{(Gas)}} + C -> 2CO} % mit Text in Formel, besser

```

ergibt:

H<sub>2</sub>O (UNSCHÖN)

H<sub>2</sub>O (OK)

H<sub>2</sub>O (BESSER)

AgCl<sub>2</sub><sup>-</sup> (UNSCHÖN)

AgCl<sub>2</sub><sup>-</sup> (BESSER)

(NH<sub>4</sub>)<sub>2</sub>S (UNSCHÖN)

(NH<sub>4</sub>)<sub>2</sub>S (BESSER)

CO<sub>2</sub>+C → 2CO (UNSCHÖN)

CO<sub>2</sub> + C → 2 CO (BESSER)

CO<sub>2</sub> + C ⇌ 2 CO

A  $\xrightarrow{+H_2O}$  B

SO<sub>4</sub><sup>2-</sup> + Ba<sup>2+</sup> → BaSO<sub>4</sub> ↓

CO<sub>2</sub>(Gas) + C → 2 CO

CO<sub>2</sub> (Gas) + C → 2 CO (BESSER)

Eine weitere Möglichkeit zur Erzeugung von Textformeln ist das Paket chemsym. Damit

ist außerdem die Darstellung von Elektronenkonfigurationen (z. B. Punkte um Elementkürzel) möglich.

## 9.6 Strukturformeln

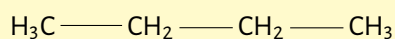
Chemische Strukturformeln können beispielsweise mit dem Paket `chemfig`<sup>14</sup> gezeichnet werden, da es Formeln direkt im PDF erzeugen kann. Ich stelle hier ein paar einfache, aus der Paket-Dokumentation entnommene Beispiele vor; für die allgemeine Syntax zur Eingabe eigener Strukturen sollte man sich im Handbuch zum Paket belesen.

---

```
1 \chemfig{H_3C-CH_2-CH_2-CH_3}
```

---

erzeugt:

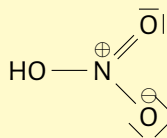


---

```
1 \chemfig{HO-[ , 0.75]\chemabove{N}{\scriptstyle\oplus}(=[1, 0.75]
↪ \lewis{02,0})-[7, 0.75]
↪ \chemabove{\lewis{157,0}}{\scriptstyle\ominus}}
```

---

erzeugt:



Wie man sieht, sind dem Aufbau der Strukturformeln keine Grenzen gesetzt.

---

<sup>14</sup>Alternative Pakete lauten: `chemtex`, `XymTeX` und `OCHEM`.

## 9.7 T<sub>E</sub>X für Mineralogen: Miller-Indizes

Mineralogen verwenden häufig die sog. Millerschen Indizes, um die Flächen eines Kristalls eindeutig zu bezeichnen. Hier bietet sich das Paket `miUler` an. Damit können dann auf ganz einfache Weise Ziffern als Millersche Indizes gekennzeichnet werden:

---

```
1 \hkl[1 4 1]
2 \hkl[-1 -1 0]
```

---

| ergibt:           |
|-------------------|
| [1 4 1]           |
| $\bar{1}\bar{1}0$ |

Entsprechend der obigen Eingaben sind statt der eckigen Klammern auch runde, dreieckige und geschweifte Klammern möglich.

## 9.8 Ordnungszahlwörter

In englischen Texten ist die Eingabe von hochgestellten Ordnungszahlwörtern möglich mit:

---

```
1 1\textsuperscript{st}, 2\textsuperscript{nd}, 3\textsuperscript{rd},
↪ 4\textsuperscript{th} usw.
```

---

| ergibt:                                                                    |
|----------------------------------------------------------------------------|
| 1 <sup>st</sup> , 2 <sup>nd</sup> , 3 <sup>rd</sup> , 4 <sup>th</sup> usw. |

Einfacher ist es allerdings mit dem `nth`-Paket, das mit der Option `super` geladen wird. Nun reicht die Eingabe der gewünschten Zahl; das Paket ergänzt automatisch das passende Kürzel (das hängt von der Grundsprache des Dokuments ab!):

- 
- 1 `\usepackage[super]{nth}`
  - 2 ...
  - 3
  - 4 `\nth{1}`, `\nth{2}`, `\nth{3}`, `\nth{4}` usw.
- 

## 9.9 Akademische Grade

Titel und akademische Grade stehen unmittelbar *vor* dem Namen

- 
- 1 Hermann Junge, Prof. (FALSCH)
  - 2 Prof.~Hermann Junge (RICHTIG)
-