

Setzen von Abbildungs- und Tabellenbeschriftungen mit dem caption-Paket*

Axel Sommerfeldt
caption@sommerfeldt.net

26. Juni 2005

Zusammenfassung

Das caption Paket bietet einem Mittel und Wege das Erscheinungsbild der Bildunterschriften und Tabellenüberschriften den eigenen Wünschen bzw. bestimmten Vorgaben anzupassen. Hierbei wurde viel Wert auf die reibungslose Zusammenarbeit und in die Integration anderer Dokumentenklassen und Pakete gelegt.¹

1 Einleitung

Mit `\caption` gesetzte Bildunterschriften und Tabellenüberschriften werden von den Standard-Dokumentenklassen eher stiefmütterlich behandelt. In der Regel schlicht als ganz normaler Absatz gesetzt, ergibt sich keine signifikante optische Abgrenzung vom eigentlichen Text, wie z.B. hier:

Abbildung 1: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

Es sollte aber eine Möglichkeit geben, diesem Umstand abzuweichen. Es wäre zum Beispiel nett, wenn man den Text der Unterschrift etwas kleiner gestalten, extra Ränder festlegen oder den Zeichensatz des Bezeichners dem der Kapitelüberschriften anpassen könnte. So in etwa:

Abbildung 2: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

Mit Hilfe dieses Paketes können Sie dies leicht bewerkstelligen; es sind viele vorgegebene Parameter einstellbar, Sie können aber auch eigene Gestaltungsmerkmale einfließen lassen.

*Dieses Paket hat die Versionsnummer v3.0g, zuletzt geändert am 2005/06/28.

¹Für die aktuelle Version 3.0 dieses Paketes wurde die Benutzerschnittstelle zusammen mit Steven D. Cochran und Frank Mittelbach komplett überarbeitet.

2 Verwendung des Paketes

`\usepackage` Mittels

```
\usepackage[Optionen]{caption}[2005/06/28]
```

in dem Vorspann des Dokumentes wird das `caption` Paket eingebunden, die Optionen legen hierbei das Aussehen der Über- und Unterschriften fest. So würde z.B.

```
\usepackage[margin=10pt,font=small,labelfont=bf]{caption}
```

zu dem obigen Ergebnis mit Rand, kleinerem Zeichensatz und fatter Bezeichnung führen.

`\captionsetup` Eine Änderung der Parameter ist auch zu einem späteren Zeitpunkt jederzeit mittels des Befehls

```
\captionsetup[Typ]{Optionen}
```

möglich. So sind z.B. die Befehlssequenzen

```
\usepackage[margin=10pt,font=small,labelfont=bf]{caption}
```

und

```
\usepackage{caption}  
\captionsetup{margin=10pt,font=small,labelfont=bf}
```

in ihrer Wirkung identisch.

Weiterhin gilt zu beachten, daß sich die Verwendung von `\captionsetup` innerhalb von Umgebungen nur auf die Umgebung selber auswirkt, nicht aber auf den Rest des Dokumentes. Möchte man also z.B. die automatische Zentrierung der Abbildungsunterschrift nur in einem konkreten Falle ausschalten, so kann dies mit

```
\begin{figure}  
...  
\captionsetup{singlelinecheck=off}  
\caption{...}  
\end{figure}
```

geschehen, ohne daß die restlichen Abbildungsunterschriften hiervon beeinträchtigt werden.

(Der optionale Parameter `<Typ>` wird in Abschnitt 4: „*Nützliches*“ behandelt.)

3 Beschreibung der Optionen

3.1 Formatierung

`format=` Eine Abbildungs- oder Tabellenbeschriftung besteht im wesentlichen aus drei Teilen: Dem Bezeichner (etwa „Abbildung 3“), dem Trenner (etwa „:““) und dem eigentlichen Text. Mittels der Option

`format=⟨Name⟩`

wird festgelegt, wie diese drei Teile zusammengesetzt werden. Für $\langle Name \rangle$ sind folgende Möglichkeiten verfügbar:²

`default` Die Beschriftung wird als normaler Abschnitt gesetzt. (Dies ist das Standardverhalten.)

`hang` Der Text wird so gesetzt, daß er an dem Bezeichner „hängt“, d.h. der Platz unter dem Bezeichner und dem Trenner bleibt leer.

Ein Beispiel: Die Angabe der Option

`format=hang`

führt zu Abbildungsunterschriften der Art

Abbildung 3: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`indention=` Bei beiden Formaten (`default` und `hang`) kann der Einzug der Beschriftung ab der zweiten Textzeile angepasst werden, dies geschieht mit Hilfe der Option

`indention=⟨Einzug⟩` ,

wobei anstelle von $\langle Einzug \rangle$ jedes beliebige feste Maß angegeben werden kann.

Drei Beispiele:

`format=default,indention=.5cm`

Abbildung 4: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`format=hang,indention=-0.5cm`

Abbildung 5: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`labelformat=` Mittels der Option

`labelformat=⟨Name⟩`

Neue Beschreibung
v3.0e wird die Zusammensetzung des Bezeichners festgelegt. Für $\langle Name \rangle$ sind folgende Möglichkeiten verfügbar:

<code>default</code>	Der Bezeichner wird wie von der verwendeten Dokumentenklasse vorgegeben gesetzt, üblicherweise ist dies der Name und die Nummer, getrennt durch ein Leerzeichen (wie <code>simple</code>). (Dies ist das Standardverhalten.)
<code>empty</code>	Der Bezeichner ist leer. Diese Option macht nur in Verbindung mit anderen Optionen (wie etwa <code>labelsep=none</code>) Sinn.
<code>simple</code>	Der Bezeichner ist aus dem Namen und der Nummer zusammengesetzt.
<code>parens</code>	Die Nummer des Bezeichners wird in runde Klammern gesetzt.

Ein Beispiel: Die Optionen

`labelformat=parens, labelsep=quad`

führen zu Abbildungsunterschriften der Art

Abbildung (6) Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`labelsep=` Mittels der Option

`labelsep=⟨Name⟩`

wird die Zusammensetzung des Trenners festgelegt. Für $\langle Name \rangle$ sind folgende Möglichkeiten verfügbar:

<code>none</code>	Der Trenner ist leer. Diese Option macht nur in Verbindung mit anderen Optionen (wie etwa <code>labelformat=empty</code>) Sinn.
<code>colon</code>	Der Trenner besteht aus einem Doppelpunkt und einem Leerzeichen. (Dies ist das Standardverhalten.)
<code>period</code>	Der Trenner besteht aus einem Punkt und einem Leerzeichen.
<code>space</code>	Der Trenner besteht lediglich aus einem einzelnen Leerzeichen.
<code>quad</code>	Der Trenner besteht aus einem <code>\quad</code> .
<code>newline</code>	Als Trenner wird ein Zeilenumbruch verwendet.

²Es gibt hier wie auch bei vielen anderen Optionen die Möglichkeit, auch eigene Formate, Zeichensätze etc. zu definieren. Wie dies geht wird in Abschnitt 5: „Eigene Erweiterungen“ dargelegt.

Zwei Beispiele:

```
labelsep=period
```

Abbildung 7. Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

```
labelsep=newline, singlelinecheck=false
```

Abbildung 8

Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

3.2 Textausrichtung

`justification=` Mittels der Option

```
justification=<Name>
```

wird die Ausrichtung des Textes festgelegt. Für *<Name>* sind folgende Möglichkeiten verfügbar:

<code>justified</code>	Der Text wird als Blocksatz gesetzt. (Dies ist das Standardverhalten.)
<code>centering</code>	Der Text wird zentriert gesetzt.
<code>centerlast</code>	Lediglich die letzte Zeile des Textes wird zentriert gesetzt.
<code>centerfirst</code>	Lediglich die erste Zeile des Textes wird zentriert gesetzt.
<code>raggedright</code>	Der Text wird linksbündig gesetzt.
<code>RaggedRight</code>	Der Text wird ebenfalls linksbündig gesetzt. Hierfür wird jedoch im Gegensatz zu <code>raggedright</code> der Befehl <code>\RaggedRight</code> des <code>ragged2e</code> -Paketes verwendet, der \LaTeX das Trennen der Worte erlaubt.
<code>raggedleft</code>	Der Text wird rechtsbündig gesetzt.

Einige Beispiele:

```
justification=centerlast
```

Abbildung 9: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`format=hang, justification=raggedright`

Abbildung 10: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`format=hang, justification=RaggedRight`

Abbildung 11: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`labelsep=newline, justification=centering`

Abbildung 12

Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`singlelinecheck=` In den Standard-Dokumentenklassen `article`, `report` und `book` sind die Abbildungs- und Tabellenbeschriftungen so realisiert, daß sie automatisch zentriert werden, wenn sie lediglich aus einer einzigen Textzeile bestehen:

Abbildung 13: Eine kurze Beschriftung.

Diesen Mechanismus übernimmt das `caption`-Paket und ignoriert damit bei solch kurzen Beschriftungen die mittels der Option `justification=` eingestellte Textausrichtung. Dieses Verhalten kann jedoch mit der Option

`singlelinecheck=<bool>`

reguliert werden. Setzt man für `<bool>` entweder `false`, `no`, `off` oder `0` ein, so wird der Zentrierungsmechanismus außer Kraft gesetzt. Die obrige, kurze Abbildungsunterschrift würde z.B. nach Angabe der Option

`singlelinecheck=false`

so aussehen:

Abbildung 13: Eine kurze Beschriftung.

Setzt man für `<bool>` hingegen `true`, `yes`, `on` oder `1` ein, so wird die automatische Zentrierung wieder eingeschaltet. (Standardmäßig ist sie eingeschaltet.)

3.3 Zeichensätze

`font=` Das `caption`-Paket kennt 3 Zeichensätze: Denjenigen für die gesamte Beschriftung
`labelfont=` (`font`), denjenigen, der lediglich auf den Bezeichner und den Trenner angewandt wird
`textfont=` (`labelfont`), sowie denjenigen, der lediglich auf den Text wirkt (`textfont`). So lassen sich die unterschiedlichen Teile der Beschriftung individuell mittels

font={⟨Zeichensatzoptionen⟩} ,
 labelfont={⟨Zeichensatzoptionen⟩} und
 textfont={⟨Zeichensatzoptionen⟩}

anpassen.

Als ⟨Zeichensatzoptionen⟩ sind eine Kombination aus folgendem erlaubt:

scriptsize	Sehr kleine Schrift
footnotesize	Fußnotengröße
small	Kleine Schrift
normalsize	Normalgroße Schrift
large	Große Schrift
Large	Größere Schrift
up	Upright Schriftart
it	<i>Italic Schriftart</i>
sl	<i>Slanted Schriftart</i>
sc	SMALL CAPS SCHRIFTART
md	Medium Schriftserie
bf	Bold Schriftserie
rm	Roman Schriftfamilie
sf	Sans Serif Schriftfamilie
tt	Typewriter Schriftfamilie

Wird lediglich eine einzelne Zeichensatzoption ausgewählt, können die geschweiften Klammern entfallen, d.h. die Optionen font={small} und font=small sind in ihrer Wirkung identisch.

Beispiele:

font={small, it}, labelfont=bf

Abbildung 14: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

font=small, labelfont=bf, textfont=it

Abbildung 15: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

3.4 Ränder und Absätze

`margin=` Für die Abbildungs- und Tabellenbeschriftungen kann *entweder* ein extra Rand *oder* eine
`width=` feste Breite festgelegt werden. Dies geschieht mit

```
margin=\Rand    oder  
width=\Breite
```

In beiden Fällen wird die Beschriftung zentriert, d.h. der linke und rechte Rand sind immer gleich groß.

Zwei Beispiele hierzu:

```
margin=10pt
```

Abbildung 16: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

```
width=.75\textwidth
```

Abbildung 17: Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

`parskip=` Diese Option wirkt auf Abbildungs- oder Tabellenbeschriftungen, die aus mehr als einem Absatz bestehen; sie legt den Abstand zwischen den Absätzen fest.

```
parskip=\Abstand zwischen Absätzen
```

Ein Beispiel:

```
margin=10pt,parskip=5pt
```

Abbildung 18: Erster Absatz der Beschriftung; dieser enthält einigen Text, so daß die Auswirkungen der Optionen deutlich werden.

Zweiter Absatz der Beschriftung; dieser enthält ebenfalls einigen Text, so daß die Auswirkungen der Optionen deutlich werden.

`hangindent=` Die Option

```
hangindent=\Einzug
```

legt einen Einzug für alle Zeilen außer der jeweils ersten des Absatzes fest. Besteht die Beschriftung lediglich aus einem einzelnen Absatz, so ist die Wirkung mit der Option `indentation=\Einzug` identisch, bei mehreren Absätzen zeigt sich jedoch der Unterschied:

```
format=hang,indention=-.5cm
```

Abbildung 19: Erster Absatz der Beschriftung; dieser enthält einigen Text, so daß die Auswirkungen der Optionen deutlich werden.
Zweiter Absatz der Beschriftung; dieser enthält ebenfalls einigen Text, so daß die Auswirkungen der Optionen deutlich werden.

```
format=hang,hangindent=-.5cm
```

Abbildung 20: Erster Absatz der Beschriftung; dieser enthält einigen Text, so daß die Auswirkungen der Optionen deutlich werden.
Zweiter Absatz der Beschriftung; dieser enthält ebenfalls einigen Text, so daß die Auswirkungen der Optionen deutlich werden.

3.5 Stile

`style=` Eine geeignete Kombination aus den bisher vorgestellten Optionen wird *Stil* genannt; dies ist in etwa mit dem Seitenstil vergleichbar, den man bekanntermaßen mit `\pagestyle` einstellen kann.

Einen vordefinierten Abbildungs- und Tabellenbeschriftungsstil kann man mit der Option

```
style=(Stil)
```

auswählen. Das `caption`-Paket definiert in der Regel lediglich den Stil `default`, der alle vorgehenden Optionen auf die Standardbelegung zurücksetzt, d.h. die Angabe der Option

```
style=default
```

entspricht den Optionen

```
format=default, labelformat=simple, labelsep=colon,  
justification=justified, font=default, labelfont=default,  
textfont=default, margin=0pt, indention=0pt, parindent=0pt  
hangindent=0pt, singlelinecheck=true
```

3.6 Abstände

`aboveskip=` Die Standard-L^AT_EX-Klassen `article`, `report` und `book` definieren zwei Abstände, die in Zusammenhang mit den Abbildungs- und Tabellenbeschriftungen gebraucht werden: `\abovecaptionskip` wird *vor* der Beschriftung angewandt und ist auf 10pt vorbelegt. `\belowcaptionskip` wird *nach* der Beschriftung angewandt und ist auf 0pt vorbelegt.

Beide Abstände lassen sich gewöhnlich mit `\setlength` ändern, aber auch mit den caption-Optionen

`aboveskip=`*(Abstand nach oben)* und
`belowskip=`*(Abstand nach unten)* .

`position=` Die sture Anwendung von `\abovecaptionskip` und `\belowcaptionskip` bei Beschriftungen birgt einen großen Nachteil: Ist die Beschriftung *über* und nicht *unter* der Abbildung oder Tabelle (wie bei Tabellen durchaus üblich), so ist die Vorbelegung dieser beiden Abstände nicht mehr sinnvoll, da hier kein Abstand zwischen der Beschriftung und der Tabelle gesetzt wird. (Wir erinnern uns: `\belowcaptionskip` ist auf 0pt, also kein Abstand, vorbelegt.)

Man vergleiche:

Tabelle 1: Eine Tabelle	A B C D
A B C D	Tabelle 2: Eine Tabelle

Mit Hilfe des Parameters `position` kann man jedoch festlegen, wie L^AT_EX die Beschriftung behandeln soll, ob als *Überschrift* oder als *Unterschrift*:

`position=top` (oder `position=above`)

gibt an, daß die Abstände so gewählt werden, daß sie für *Überschriften* sinnvoll gewählt sind und

`position=bottom` (oder `position=below`)

gibt an, daß die Abstände so gewählt werden, daß sie für *Überschriften* sinnvoll gewählt sind. (Dies ist die Standardeinstellung, allerdings nicht für `longtables`.)

Fügt man nun ein `\captionsetup{position=top}` in die linke Tabelle vor dem `\caption` Befehl ein, so ergeben sich im Vergleich zwischen *Überschrift* und *Unterschrift* stimmige Abstände:

Tabelle 3: Eine Tabelle	A B C D
A B C D	Tabelle 4: Eine Tabelle

(Technisch ist dies so realisiert, daß bei *Überschriften* die Abstände `\abovecaptionskip` und `\belowcaptionskip` vertauscht werden, so daß `\abovecaptionskip` immer derjenige Abstand ist, der zwischen Tabelle und Beschriftung zur Anwendung kommt.)

`tableposition=` Sinnvoll läßt sich diese Option insbesondere zusammen mit dem optionalen Parameter bei `\captionsetup` anwenden (siehe auch Abschnitt 4: „*Nützliches*“):

`\captionsetup[table]{position=top}`

Neues Feature
v3.0a

bewirkt, daß bei allen Tabellen die Beschriftung bzgl. der Abstände als Überschrift behandelt wird. Da dies eine sehr gebräuchliche Einstellung ist, stellt das caption-Paket diesen Befehl auch als abkürzende Option für `\usepackage` zur Verfügung:

```
\usepackage[... ,tableposition=top]{caption}
```

entspricht

```
\usepackage[...]{caption}  
\captionsetup[table]{position=top}
```

4 Nützliches

`\caption` Der Befehl

```
\caption[<Kurzform für das Verzeichnis>]{<Beschriftung>}
```

erzeugt eine Über- bzw. Unterschrift innerhalb einer gleitenden Umgebung wie `figure` oder `table`. Dies ist an sich nichts neues, neu ist allerdings, daß kein Eintrag ins Abbildungs- oder Tabellenverzeichnis vorgenommen wird, wenn eine leere Kurzform angegeben wird, etwa so:

```
\caption[] {Dies ist eine Abbildung, die nicht ins  
Abbildungsverzeichnis aufgenommen wird}
```

`\caption*` Das `longtable`-Paket definiert zusätzlich zum Befehl `\caption` auch den Befehl `\caption*`, der eine Beschriftung ohne Bezeichner und ohne Eintrag ins Tabellenverzeichnis erzeugt. So erzeugt z.B. der Code

```
\begin{longtable}{cc}  
  \caption*{Eine Tabelle} \\  
  A & B \\  
  C & D \\  
\end{longtable}
```

diese Tabelle:

Eine Tabelle

A	B
C	D

Das `caption`-Paket bietet diesen Mechanismus auch für normale Abbildungs- und Tabellenbeschriftungen an. Ein Beispiel:

```
\begin{table}
```

```

\caption*{Eine Tabelle}
\begin{tabular}{cc}
A & B \\
C & D
\end{longtable}
\end{table}

```

`\captionof` `\captionof*` Der Befehl `\caption` funktioniert in der Regel nur innerhalb von gleitenden Umgebungen, manchmal möchte man ihn jedoch auch anderweitig anwenden, etwa um eine Abbildung in eine nicht-gleitende Umgebung wie `minipage` zu setzen.

Hierfür stellt das `caption`-Paket den Befehl

```
\captionof{<Umgebungstyp>}[<Kurzform>]{<Langform>}
```

zur Verfügung. Die Angabe des Umgebungstypen ist hierbei notwendig, damit der gewünschte Bezeichner gewählt wird und der Eintrag in das richtige Verzeichnis vorgenommen wird. Ein Beispiel:

```

\captionof{figure}{Abbildung}
\captionof{table}{Tabelle}

```

führt zu folgendem Ergebnis:

Abbildung 21: Abbildung

Tabelle 6: Tabelle

Analog zu `\caption*` gibt es auch den Befehl `\captionof*` für Beschriftungen ohne Bezeichner und ohne Verzeichniseintrag.

Beide Befehle, sowohl `\captionof` als auch `\captionof*`, sollten nur *innerhalb* von Umgebungen (wie `minipage` oder `\parbox`) angewandt werden. Zum einen kann es sonst passieren, daß \LaTeX zwischen dem Inhalt und der Beschreibung einen Seitenumbruch setzt; zum anderen kann es bei Mißachtung gar zu seltsamen Effekten (wie falsche Abstände um die Beschriftungen herum) kommen!

`\ContinuedFloat` Manchmal möchte man Abbildungen oder Tabellen aufteilen, jedoch ohne den einzelnen Teilen eine eigene Abbildungs- oder Tabellenummer zu geben. Hierfür stellt das `caption`-Paket den Befehl

```
\ContinuedFloat
```

zur Verfügung, der gleich als erstes innerhalb der folgenden Teile angewandt werden sollte. Er verhindert, daß die Zählung fortgeführt wird; eine Abbildung oder Tabelle, die ein `\ContinuedFloat` enthält, erhält also die gleiche Nummer wie die vorherige Abbildung oder Tabelle.

Ein Beispiel:

```

\begin{table}
\caption{Eine Tabelle.}
...
\end{table}
...
\begin{table}\ContinuedFloat
\caption{Eine Tabelle. (fortgef"uhrt)}
...
\end{table}

```

ergibt als Ergebnis:

Tabelle 7: Eine Tabelle.

...

Tabelle 7: Eine Tabelle. (fortgeführt)

`\captionsetup` Den Befehl `\captionsetup` haben wir ja schon im Abschnitt 2: „*Verwendung des Paketes*“ kennengelernt, uns dort allerdings die Bedeutung des optionalen Parameters $\langle Typ \rangle$ aufgespart.

Wir erinnern uns, die Syntax des Befehls lautet

```
\captionsetup[\langle Typ \rangle]{\langle Optionen \rangle}
```

Wird hierbei ein $\langle Typ \rangle$ angegeben, so werden die Optionen nicht unmittelbar umgesetzt, sondern werden lediglich vermerkt und kommen erst dann zum Einsatz, wenn eine Über- bzw. Unterschrift innerhalb des passenden gleitenden Umgebungstyps gesetzt wird. So wirkt sich z.B. die Angabe

```
\captionsetup[figure]{\langle Optionen \rangle}
```

lediglich auf die Unterschriften aus, die innerhalb einer `figure`-Umgebung gesetzt werden.

Ein Beispiel:

```

\captionsetup{font=small}
\captionsetup[figure]{labelfont=bf}

```

führt zu Abbildungs- und Tabellenunterschriften der Art:

Abbildung 22: Eine Abbildung

Tabelle 8: Eine Tabelle

Wie man sieht, führt das `\captionsetup[figure]{labelfont=bf}` lediglich dazu, daß Abbildungsunterschriften mit fettem Bezeichner gesetzt werden, alle anderen Unter- bzw. Überschriften werden hiervon nicht beeinflusst.

`\clearcaptionsetup` Um diese vermerkten, typbezogenen Parameter aus dem Gedächtnis von \LaTeX zu löschen, gibt es den Befehl

```
\clearcaptionsetup{<Typ>} .
```

`\clearcaptionsetup{figure}` würde z.B. die obrige Sonderbehandlung der Abbildungsunterschriften wieder aufheben:

Abbildung 23: Eine Abbildung

Tabelle 9: Eine Tabelle

Als Umgebungstypen mit Unter- bzw. Überschriften gibt es in der Regel nur zwei: `figure` und `table`. Wie wir jedoch später sehen werden, kommen durch die Verwendung spezieller \LaTeX -Pakete (wie etwa das `float`-Paket) ggf. weitere Typen hinzu, deren Beschriftungen ebenfalls derart individuell angepasst werden können.

5 Eigene Erweiterungen

Wem die vorhandenen Formate, Trenner, Textausrichtungen, Zeichensätze und Stile nicht ausreichen, der hat die Möglichkeit, sich eigene zu definieren. Hierzu gibt es eine Reihe von Befehlen, die in dem Vorspann des Dokumentes (das ist der Teil vor `\begin{document}`) zum Einsatz kommen.

`\DeclareCaptionFormat` Eigene Formate können mit dem Befehl

```
\DeclareCaptionFormat{<Name>}{<Code mit #1, #2 und #3>}
```

definiert werden. Für `#1` wird später der Bezeichner, für `#2` der Trenner und für `#3` der Text eingesetzt. So ist z.B. das Standardformat `default`, welches die Beschriftung als gewöhnlichen Absatz formatiert, folgendermaßen in `caption.sty` definiert:

```
\DeclareCaptionFormat{default}{#1#2#3\par}
```

`\DeclareCaptionLabelFormat` Ähnlich können auch eigene Bezeichnerformate definiert werden:

```
\DeclareCaptionLabelFormat{<Name>}{<Code mit #1 und #2>}
```

Bei den Bezeichnerformaten wird hierbei für `#1` der Name (also z.B. „Abbildung“), für `#2` die Nummer (also z.B. „12“) eingesetzt.

`\bothIfFirst` `\bothIfSecond` Hierbei gibt es eine Besonderheit zu beachten: Wird das Bezeichnerformat auch in Verbindung mit dem `subfig`-Paket[10] verwendet, so kann der Bezeichnername (also `#1`) auch leer sein. Um dies flexibel handhaben zu können, stellt das `caption`-Paket die Befehle

```
\bothIfFirst{<Erstes Argument>}{<Zweites Argument>} und  
\bothIfSecond{<Erstes Argument>}{<Zweites Argument>}
```

zur Verfügung. `\bothIfFirst` testet, ob das erste Argument leer ist, `\bothIfSecond`, ob das zweite Argument leer ist. Nur wenn dies nicht der Fall ist, werden beide Argumente ausgegeben, ansonsten werden beide unterdrückt.

So ist z.B. das Standard-Bezeichnerformat `simple` nicht, wie man naiverweise annehmen könnte, als

```
\DeclareCaptionLabelFormat{simple}{#1 #2}
```

definiert, weil dies zu einem störendem führenden Leerzeichen führen würde, wenn #1 leer ist. Stattdessen findet sich in `caption.sty` folgende Definition, die sowohl mit `\caption` als auch mit `\subfloat` harmoniert:

```
\DeclareCaptionLabelFormat{simple}{\bothIfFirst{#1}{ }#2}
```

d.h. das Leerzeichen kommt nur dann zum Einsatz, wenn #1 nicht leer ist.

`\DeclareCaptionLabelSeparator` Eigene Trenner werden mit

```
\DeclareCaptionLabelSeparator{<Name>}{<Code>}
```

definiert. Auch hier wieder als einfaches Beispiel die Definition des Standard-Trenners aus `caption.sty`:

```
\DeclareCaptionLabelSeparator{colon}{: }
```

`\DeclareCaptionJustification` Eigene Textausrichtungen können mit

```
\DeclareCaptionJustification{<Name>}{<Code>}
```

definiert werden. Der `<Code>` wird dann der Beschriftung vorangestellt, so führt z.B. die Verwendung der Ausrichtung

```
\DeclareCaptionJustification{raggedright}{\raggedright}
```

dazu, daß alle Zeilen der Beschriftung linksbündig ausgegeben werden.

`\DeclareCaptionFont` Eigene Zeichensatzoptionen können mit

```
\DeclareCaptionFont{<Name>}{<Code>}
```

definiert werden. So definiert z.B. das `caption`-Paket die Optionen `small` und `bf` folgendermaßen:

```
\DeclareCaptionFont{small}{\small}
```

```
\DeclareCaptionFont{bf}{\bfseries}
```

`\DeclareCaptionStyle` Zu guter letzt noch die Definition eigener Stile. Stile sind einfach eine Ansammlung von geeigneten Einstellungen, die unter einem eigenen Namen zusammengefasst werden und mit der Paketoption `style=<Name>` ausgewählt werden können. Sie werden wie folgt definiert:

```
\DeclareCaptionStyle{<Name>}[<zusätzliche Optionen>]{<Optionen>}
```

Hierbei ist zu beachten, daß die `<Optionen>` immer auf den Standardeinstellungen basieren (siehe auch Abschnitt 3.5: „Stile“), es brauchen also nur davon abweichende Optionen angegeben werden.

Sind `<zusätzliche Optionen>` angegeben, so kommen diese automatisch zusätzlich zum Einsatz, sofern die Beschreibung in eine einzelne Zeile passt und diese Abfrage nicht mit `singlelinecheck=off` ausgeschaltet wurde.

Als Beispiel muß mal wieder eine einfache Definition aus diesem Paket herhalten:

```
\DeclareCaptionStyle{default}[justification=centering]{}</pre></div><div data-bbox="526 879 550 893" data-label="Page-Footer">

15


```

5.1 Beispiele

Möchte man als Trenner einen Punkt *und* einen Zeilenumbruch haben, so ließe sich das wie folgt realisieren:

```
\DeclareCaptionLabelSeparator{period-newline}{. \\\}
```

Wählt man diesen Trenner mit `\captionsetup{labelsep=period-newline}` aus, so ergeben sich Beschriftungen der Art

Abbildung 24.

Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

Für kurze Beschriftungen, die in eine Zeile passen, mag dieses Erscheinungsbild jedoch störend sein, selbst wenn die automatische Zentrierung (mit `singlelinecheck=off`) ausgeschaltet ist:

Abbildung 25.

Eine Abbildung.

Abhilfe schafft ein eigener Stil, der bei solchen Beschriftungen einen anderen Trenner ohne Zeilenumbruch auswählt:

```
\DeclareCaptionStyle{period-newline}%  
[labelsep=period]{labelsep=period-newline}
```

Abbildung 25. Eine Abbildung.

Möchte man die automatische Zentrierung ebenfalls implementieren, so wäre

```
\DeclareCaptionStyle{period-newline}%  
[labelsep=period, justification=centering]%  
{labelsep=period-newline}
```

eine geeignete Definition:

Abbildung 25. Eine Abbildung.

Leicht abgewandelt ergibt sich auch bei längeren Beschriftungen eine Zentrierung:

```
\DeclareCaptionStyle{period-newline}%  
[labelsep=period]%  
{labelsep=period-newline, justification=centering}
```

Abbildung 26.

Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

Ein anderes Beispiel: Die Beschriftungen sollen wie folgt aussehen:

Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)
(Abbildung 27)

Dies ließe sich z.B. wie folgt realisieren:

```
\DeclareCaptionFormat{reverse}{#3#2#1}
\DeclareCaptionLabelFormat{fullparens}{(\bothIfFirst{#1}{ }#2)}
\DeclareCaptionLabelSeparator{fill}{\hfill}
\captionsetup{format=reverse,labelformat=fullparens,
              labelsep=fill,font=small,labelfont=it}
```

Ein weiteres Beispiel: Der Bezeichner soll in den linken Rand verlagert werden, so daß die komplette Absatzbreite der Beschriftung selber zugute kommt:

```
\DeclareCaptionFormat{llap}{\llap{#1#2}#3\par}
\captionsetup{format=llap,singlelinecheck=no,labelsep=quad}
```

Das Ergebnis wären Beschriftungen wie diese:

Abbildung 28 Die auf die Rotationsfrequenz des Innenzylinders normierten Eigenfrequenzen der gefundenen Grundmoden der Taylor-Strömung für $\eta = 0.5$. (Die azimutale Wellenzahl ist mit m bezeichnet.)

6 Verwendung mit speziellen Dokumentenklassen

Neue Beschreibung v3.0d Das `caption`-Paket ist auf die Verwendung mit den Standard-Dokumentenklassen `article`, `report` und `book` ausgelegt.

Möchte man das `caption`-Paket mit den KOMA-Script-Klassen oder der `memoir`-Klasse verwenden, so ist sorgfältig abzuwägen, ob die in diesen Klassen reichlich enthaltenden Möglichkeiten zur Anpassung der Beschriftungen nicht ausreichend sind. Denn das `caption`-Paket funktioniert zwar auch mit den KOMA-Script-Klassen und der `memoir`-Klasse, allerdings sollte darauf geachtet werden, daß die verschiedenen Anpassungsmöglichkeiten nicht im Dokument vermischt werden. Ist das `caption`-Paket erst einmal geladen, verlieren die Optionen der Dokumentenklassen wie `tablecaptionabove` und Befehle wie `\captionabove`, `\captionbelow`, `\captionformat`, `\figureformat`, `\tableformat`, `\setcapindent`, `\setcaphanging`, `\captionstyle` usw. ihre Wirkung.

Das `caption`-Paket sollte zwar auch mit anderen als den oben erwähnten Dokumentenklassen zusammenarbeiten, ist jedoch nicht uneingeschränkt empfehlenswert, da es zu unerwünschten Layout-Änderungen, sonstigen Nebenwirkungen oder gar Fehlfunktionen kommen kann. (Zukünftige Versionen des `caption`-Paketes werden allerdings mehr Dokumentenklassen unterstützen als dies derzeit der Fall ist.)

7 Interaktion mit anderen Paketen

Das `caption`-Paket enthält spezielle Anpassungen an andere Pakete, damit die Beschriftungen dort ebenfalls genauso ausgegeben werden, wie man es mit den oben genannten Optionen eingestellt hat. Im einzelnen sind dies die folgenden Pakete:

<code>float</code>	Erlaubt die Definition eigener gleitenden Umgebungsstile
<code>hyperref</code>	Setzt Hyperlinks
<code>hyperc</code>	Korrigiert die von <code>hyperref</code> gesetzten Anker bei Beschriftungen
<code>listings</code>	Setzt Quelltexte diverser Programmiersprachen
<code>longtable</code>	Setzt Tabellen, die sich über mehrere Seiten erstrecken können
<code>rotating</code>	Unterstützt rotierende Abbildungen und Tabellen
<code>sidecap</code>	Erlaubt das Setzen von Beschriftungen <i>neben</i> Abbildungen
<code>supertabular</code>	Setzt Tabellen, die sich über mehrere Seiten erstrecken können

Neues Feature
v3.0b

Wird eines dieser Pakete zusammen mit dem `caption`-Paket verwendet, steht einem zusätzlich die Möglichkeit bereit, mittels

```
\captionsetup[Umgebung]{Optionen}
```

Optionen festzulegen, die lediglich für diese Umgebung gelten. So bewirkt z.B.

```
\captionsetup[lstlisting]{labelfont=bf} ,
```

daß Beschriftungen innerhalb der `lstlisting`-Umgebung automatisch mit fettem Bezeichner gesetzt werden. (Dies funktioniert allerdings nicht mit den `sideways`-Umgebungen, die vom `rotating`-Paket angeboten werden.)

Ist die Unterstützung eines dieser Pakete im speziellen Falle nicht erwünscht, so kann dies durch Angabe der Option

```
\usepackage[... , Paketname=no]{caption}
```

erreicht werden, etwa `float=no`, wenn die Unterstützung des `float`-Paketes nicht erwünscht wird. (Hinweis: Diese Optionen können nur innerhalb des `\usepackage`-Befehls angegeben werden, also insbesondere *nicht* nachträglich mit `\captionsetup`.)

Nähere Informationen zu den einzelnen Paketen entnehmen Sie bitte der dazugehörigen Anleitung oder dem `LATEX`-Begleiter[1].

7.1 Das float-Paket

Ein sehr nützliches Feature stellt das `float`-Paket[2] mit dem Platzierungsparameter `[H]` bereit: Anders als beim Parameter `h`, den `LATEX` bereitstellt und der lediglich eine Empfehlung darstellt, die Abbildung oder Tabelle nach Möglichkeit „hier“ zu setzen, ist das `H` viel kompromißloser, denn es erzwingt das Setzen genau an diesem Ort und nirgendwo anders.

Weiterhin definiert es für gleitende Umgebungen die Stile `plain`, `plaintop`, `ruled` und `boxed`. Für die vorhandenen Umgebungen `figure` und `table` kann hierbei

der Stil geändert werden, ferner können eigene Umgebungstypen definiert werden, die ggf. ein eigenes Verzeichnis bekommen.

Bzgl. der Abstände ist zu beachten, daß bei Umgebungen des Stils `plain`, `plaintop` und `boxed` lediglich `\abovecaptionskip` (und kein `\belowcaptionskip`) zur Anwendung kommt. Ferner wird dieser Abstand vom `float`-Paket gesetzt und nicht vom `caption`-Paket; das hat für Sie als Anwender zur Folge, daß dieser Abstand nicht innerhalb der Umgebung selbst mit `\captionsetup` verändert werden kann, ferner ist die Option `position=` bei diesen Umgebungen wirkungslos.

Beim Stil `boxed` ist der Abstand mit

```
\captionsetup[boxed]{skip=2pt}
```

voreingestellt, bei `plain` und `plaintop` wird die globale Einstellung verwendet, sofern keine lokale definiert wurde.

Kommt das `float`-Paket zusammen mit dem `caption`-Paket zum Einsatz, so wird automatisch der Beschriftungsstil `ruled` definiert, der das Aussehen der Beschriftungen innerhalb von Abbildungen des Umgebungsstils `ruled` regelt:

```
\DeclareCaptionStyle{ruled}{labelfont=bf,labelsep=space}
```

Für den Anwender bedeutet dies, daß allgemein getätigte Einstellungen an den Beschriftungen hier unwirksam sind; stattdessen werden die Beschriftungen genau so gesetzt, wie vom `float`-Paket stilistisch vorgegeben, nämlich so:

Programm 7.1 Das erste Programm. Dies hat nichts mit dem Paket an sich zu tun, dient aber als Beispiel. Man beachte den `ruled` Stil.

```
#include <stdio.h>

int main(int argc, char **argv)
{
    for (int i = 0; i < argc; ++i)
        printf("argv[%d] = %s\n", i, argv[i]);
    return 0;
}
```

Möchte man das Aussehen dieser Beschriftungen ändern, ist der Stil `ruled` mittels

```
\DeclareCaptionStyle{ruled}{(Optionen)}
```

geeignet zu definieren.

Dieser Mechanismus funktioniert auch bei allen anderen Umgebungsstilen, d.h. auch bei Umgebungen des Types `plain`, `plaintop` und `boxed` kann ein geeigneter Beschriftungsstil mit gleichem Namen definiert werden, der dann automatisch bei gleitenden Umgebungen dieses Types zum Einsatz kommt.

7.2 Das listings-Paket

Neue Beschreibung
v3.0b

Das listings-Paket[6] erlaubt die Einbindung von Quelltexten diverser Programmiersprachen.

Achtung: Die Zusammenarbeit klappt erst ab der Version 1.2 des listings-Paketes. Ältere Versionen verursachen eine Fehlermeldung!

7.3 Das longtable-Paket

Das longtable-Paket[7] definiert die Umgebung `longtable`, die sich wie eine Tabelle anwenden läßt, aber im Gegensatz zu `tabular` Seitenumbrüche innerhalb der Tabelle gestattet.

7.4 Das rotating-Paket

Das rotating-Paket[8] stellt u.a. die gleitenden Umgebungen `sidewaysfigure` und `sidewaystable` bereit, die im Gegensatz zu `figure` und `table` um 90 Grad gedreht werden und immer eine ganze Seite einnehmen.

7.5 Das sidecap-Paket

Neue Beschreibung
v3.0b

Das sidecap-Paket[9] definiert die Umgebungen `SCfigure` und `SCtable`, die analog zu `figure` und `table` funktionieren, aber Beschriftungen *neben* der Abbildung oder Tabelle erlauben.

Bei der Verwendung des sidecap-Paketes ist zu beachten, daß eine dort angegebene Option `raggedright`, `raggedleft` oder `ragged` den Parameter `justification=` des caption-Paketes für die betreffenden Umgebungen überschreibt, d.h. bei Beschriftungen *neben* der Abbildung oder Tabelle erhält die Option des sidecap-Paketes den Vorzug.

`listof=`

Ferner liegt es im Design des sidecap-Paketes begründet, daß das Unterdrücken des Verzeichniseintrages mit `\caption[...]` hier nicht klappt. Als Alternative kann in diesen Fällen die Unterdrückung mittels `\captionsetup{listof=false}` innerhalb der Abbildung oder Tabelle geschehen.

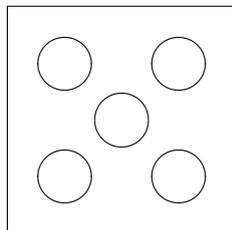


Abbildung 29: Ein kleines Beispiel mit Beschriftung neben der Abbildung.

7.6 Das supertabular-Paket

Das `supertabular`-Paket[11] definiert, ähnlich wie das `longtable`-Paket, eine Umgebung `supertabular`, die sich ebenfalls wie eine Tabelle anwenden läßt, aber im Gegensatz zu `tabular` Seitenumbrüche innerhalb der Tabelle gestattet.

Eine ausführliche Behandlung der Unterschiede zwischen den beiden Paketen `longtable` und `supertabular` ist im Buch „Der \LaTeX -Begleiter“ [1] zu finden.

7.7 Bekannte Inkompatibilitäten

Neue Beschreibung
v3.0b Die Verwendung des `caption`-Paketes in Verbindung mit folgenden Paketen ist nicht empfehlenswert, da es zu unerwünschten Seiteneffekten oder gar Fehlern kommen kann:

`ccaption`, `hvfloating`, `nonfloating`

8 Kompatibilität zu älteren Versionen

8.1 caption Version 1.x

Diese Version des `caption`-Paketes ist weitgehend kompatibel zu älteren Versionen des Paket; alte Dokumente sollten sich also in der Regel ohne Probleme weiterhin übersetzen lassen. Jedoch ist zu beachten, daß eine Mischung aus alten Befehlen und neueren Optionen bzw. Befehlen nicht unterstützt wird bzw. zu unerwünschten Nebeneffekten führen kann.

Hier eine kurze Übersicht über die alten Optionen und Befehle und durch was sie in dieser Version des `caption`-Paketes ersetzt wurden:

<code>caption 1.x</code>	<code>caption 3.x</code>
<code>normal</code>	<code>format=default</code>
<code>hang</code>	<code>format=hang</code>
<code>isu</code>	<code>format=hang</code>
<code>center</code>	<code>justification=centering</code>
<code>centerlast</code>	<code>justification=centerlast</code>
<code>anne</code>	<code>justification=centerlast</code>
<code>nooneline</code>	<code>singlelinecheck=off</code>
<code>scriptsize</code>	<code>font=scriptsize</code>
<code>footnotesize</code>	<code>font=footnotesize</code>
<code>small</code>	<code>font=small</code>
<code>normalsize</code>	<code>font=normalsize</code>
<code>large</code>	<code>font=large</code>
<code>Large</code>	<code>font=Large</code>
<code>up</code>	<code>labelfont=up</code>
<code>it</code>	<code>labelfont=it</code>
<code>sl</code>	<code>labelfont=sl</code>
<code>sc</code>	<code>labelfont=sc</code>
<code>md</code>	<code>labelfont=md</code>
<code>bf</code>	<code>labelfont=bf</code>
<code>rm</code>	<code>labelfont=rm</code>

caption 1.x	caption 3.x
sf	labelfont=sf
tt	labelfont=tt
\setlength{\captionmargin}	margin= <i>(Rand)</i>
\renewcommand{\captionfont}	\DeclareCaptionFont + \captionsetup{font= <i>(Name)</i> }
\renewcommand{\captionsize}	\DeclareCaptionFont + \captionsetup{font= <i>(Name)</i> }
\renewcommand{\captionlabelfont}	\DeclareCaptionLabelFont + \captionsetup{labelfont= <i>(Name)</i> }

8.2 caption2 Version 2.x

Die Pakete `caption` und `caption2` sind vom internen Konzept her zu unterschiedlich, um hier eine Kompatibilität gewährleisten zu können. Daher liegt diesem Paket weiterhin ein `caption2.sty` bei, so daß ältere Dokumente, die `caption2.sty` verwenden, weiterhin übersetzt werden können. Neue Dokumente sollten jedoch auf dem aktuellen `caption`-Paket aufgesetzt werden, wie es in diesem Dokument beschrieben ist.

9 Weiterführende Dokumente

Folgende Dokumente möchte ich an dieser Stelle jedem ans Herz legen:

- Die DANTE-FAQ:

<http://www.dante.de/faq/de-tex-faq/>

- „*Gleitobjekte – die richtige Schmierung*“ von Axel Reichert erläutert den Umgang mit gleitenden Umgebungen und ist hier im Netz zu finden:

<ftp://ftp.dante.de/pub/tex/info/german/gleitobjekte/>

- `epslatex` von Keith Reckdahl enthält viele nützliche Tips im Zusammenhang mit der Einbindung von Graphiken in $\LaTeX 2_{\epsilon}$. Das Dokument ist unter

<ftp://ftp.dante.de/pub/tex/info/>

als `epslatex.ps` bzw. `epslatex.pdf` zu finden.

10 Danksagungen

Von ganzem Herzen danke ich Katja Melzner, Steven D. Cochran, Frank Mittelbach, David Carlisle, Carsten Hinz und Olga Lapko.

Weiterhin möchte ich mich herzlich bei Harald Harders, Peter Löffler, Peng Yu, Alexander Zimmermann, Matthias Pospiech, Jürgen Wieferink, Christoph Bartoschek, Uwe Stöhr und Ralf Stubner für ihre Hilfe beim stetigen Verbessern dieses Paketes bedanken.

Literatur

- [1] Frank Mittelbach and Michel Goossens: *The L^AT_EX Companion (2nd. Ed.)*, Addison-Wesley, 2004.
- [2] Anselm Lingnau: *An Improved Environment for Floats*, 2001/11/08
- [3] Olga Lapko: *The floatrow package documentation*, 2005/05/22
- [4] Sebastian Rahtz: *Hypertext marks in L^AT_EX*, 2003/11/30
- [5] Heiko Oberdiek: *The hypcap package – Adjusting anchors of captions* 2001/08/27
- [6] Carsten Heinz: *The Listings Package*, 2004/02/13
- [7] David Carlisle: *The longtable package*, 2000/10/22
- [8] Sebastian Rahtz and Leonor Barroca: *A style option for rotated objects in L^AT_EX*, 1997/09/26
- [9] Rolf Niepraschk und Hubert Gäßlein: *The sidecap package*, 2003/06/06
- [10] Steven D. Cochran: *The subfig package*, 2003/12/01
- [11] Johannes Braams und Theo Jurriens: *The supertabular environment*, 2002/07/19