Gleitobjekte – die richtige Schmierung

Axel Reichert
reich@mpie-duesseldorf.mpg.de

1997-10-10

Zusammenfassung

Dieses Tutorium erläutert den Umgang mit Gleitobjekten. Die Plazierung von Floats und die Erstellung einer Legende werden behandelt. Die Einbettung und Beschriftung von Abbildungen wird erklärt. Die Gestaltung von Tabellen und umfangreichem Zahlenmaterial wird dargestellt.

Im zweiten Teil werden Hinweise zur Beschaffung und Installation zusätzlicher LaTeX 2ε-Pakete gegeben und weitere Informationsquellen genannt, Motto: Hilfe zur Selbsthilfe.

Das Tutorium richtet sich an neugierige LaTeX-Anwender mit Grundkenntnissen, aber vielleicht wird auch ein Guru eine Alternative zum selbstgebastelten Hack kennenlernen.

Inhaltsverzeichnis

1 Gleitobjekte – die richtige Schmierung

1.1 Allgemeines

1.1.1 Platzverweise

Optionale Argumente. Sowohl die `figure`- als auch die `table`-Umgebung besitzen ein optionales Argument, mit dem die erlaubte Plazierung angegeben werden kann. Folgende Argumente sind definiert:

h Hier. Bei den `*`-Formen der Umgebungen nicht erlaubt.

t Oben. Wenn der Text, der bei einer `h`-Plazierung *vor* diesem Gleitobjekt auf der laufenden Seite stünde, auch noch *nach* dem Gleitobjekt auf die Seite paßt, erfolgt eine Plazierung oben auf *dieser* Seite, sonst oben auf der *nächsten* Seite, und die laufende Seite wird mit dem auf das Gleitobjekt folgenden Text aufgefüllt.

b Unten. Wenn dieses Gleitobjekt noch unten auf die laufende Seite paßt, erfolgt die Plazierung dort, der Platz bis dorthin wird mit dem auf das Gleitobjekt folgenden Text aufgefüllt. Ist die laufende Seite dagegen schon zu weit mit Text gefüllt, wird das Gleitobjekt unten auf der nächsten Seite plaziert. Dieses Argument ist bei den `*`-Formen der Umgebungen nicht erlaubt.

p Das Gleitobjekt wird auf einer eigenen Seite ausgegeben, die durchaus auch mehrere Gleitobjekte enthalten kann.

! Die im nächsten Absatz besprochenen Plazierungsparameter werden ignoriert.

Die Voreinstellung für das optionale Argument ist `[tbp]`. In der Regel sollte man es dabei belassen, da Gleitobjekte als getrennte Textebene gedacht sind, eine `h`-Plazierung aber den Lesefluß stört.

Plazierungsparameter. Das Verhalten von LaTeX bei der Plazierung von Gleitobjekten wird durch Plazierungsparameter gesteuert. Folgende Parameter sind definiert:

\floatpagefraction Minimaler Anteil einer Seite, der bei einer `p`-Plazierung eines oder mehrerer Gleitobjekte erreicht werden muß. Voreinstellung: 0,5.

\topfraction Maximaler Anteil einer Seite, der bei einer `t`-Plazierung eines oder mehrerer Gleitobjekte erreicht werden darf. Voreinstellung: 0,7.

\bottomfraction Maximaler Anteil einer Seite, der bei einer `b`-Plazierung eines oder mehrerer Gleitobjekte erreicht werden darf. Voreinstellung: 0,3.

\textfraction Minimaler Textanteil einer Seite, der bei einer `h`-, `t`- oder `b`-Plazierung eines Gleitobjektes erreicht werden muß. Voreinstellung: 0,2.

Einer der Werte `\topfraction` oder `\bottomfraction` sollte größer als `\floatpagefraction` sein. Diese Befehle können über `\renewcommand` geändert werden. Weitere Parameter sind:

topnumber Maximale Anzahl der Gleitobjekte mit `t`-Plazierung auf der Seite. Voreinstellung: 2.

bottomnumber Maximale Anzahl der Gleitobjekte mit `b`-Plazierung auf der Seite. Voreinstellung: 1.

totalnumber Maximale Anzahl der Gleitobjekte auf der gesamten Seite. Voreinstellung: 3.

Diese Zähler können über `\setcounter` geändert werden.

flafter.sty. Dieses Paket erzwingt die Plazierung eines Gleitobjektes *nach* seiner Deklaration. Damit kann das Gleitobjekt frühestens an der Stelle seines Aufrufs erscheinen, falls eine `h`-Plazierung möglich ist, ansonsten unten auf der laufenden Seite oder oben auf der nächsten Seite.

afterpage.sty. Mit diesem Paket kann die Ausführung eines Befehls verzögert werden, so daß sie erst nach der Ausgabe der laufenden Seite erfolgt. Mit

```
\afterpage{\clearpage}
```

wird z. B. die laufende Seite noch mit nachfolgendem Text aufgefüllt, bevor der `\clearpage`-Befehl die unmittelbare Ausgabe aller anstehenden Gleitobjekte erzwingt.

Besonders praktisch ist diese Möglichkeit bei `longtable`-Umgebungen (siehe Abschnitt 1.3.2 auf Seite 13), deren Ausgabe auf diese Weise erst oben auf der nächsten Seite beginnt. So wird vermieden, daß sich unten auf der laufenden Seite nur wenige Tabellenzeilen befinden.

endfloat.sty. Dieses Paket schiebt alle Gleitobjekte an das Ende des Dokumentes und erzeugt dabei auf Wunsch Abbildungs- und Tabellenverzeichnisse sowie Markierungen im Text an den Stellen, wo die Gleitobjekte deklariert wurden.

Dies ist nicht nur für Veröffentlichungen praktisch, sondern auch für Texte mit einem sehr hohen Gleitobjektanteil. In solchen Fällen lassen sich die vielen Gleitobjekte nicht mehr sinnvoll im Text anordnen, so daß eine Plazierung am Ende des Textes besser ist. Dieses Verhalten erreicht man etwa mit

```
\usepackage[nolists,nomarkers]{endfloat}
\renewcommand{\efloatseparator}{}
```

Auf diese Weise werden keine Verzeichnisse und keine Markierungen im Text erzeugt. Außerdem wird der Seitenumbruch zwischen den Gleitobjekten (Voreinstellung) unterdrückt.

lscape.sty. Dieses Paket stellt eine `landscape`-Umgebung zur Verfügung, in der die Textseite (außer den Kopf- und Fußzeilen) um 90° gegen den Uhrzeigersinn gedreht wird. Es gehört zum `graphics`-Bündel (siehe Abschnitt 1.2.1 auf Seite 8) und erfordert/erkennt die gleichen Optionen.

varioref.sty. Dieses Paket stellt verbesserte Querverweise zur Verfügung. Mit

```
\vref{marke}
```

wird zusätzlich zur Nummer ein Hinweis auf die Seite erzeugt: Befindet sich die Marke auf der nächsten oder vorigen Seite, werden entsprechende Texte ausgegeben, ist die Marke mehr als eine Seite von dem Querverweis entfernt, wird die Seitenzahl explizit angegeben. Stehen Marke und Querverweis auf einer Seite, wird die zusätzliche Information unterdrückt. Das Paket erkennt diverse Sprachen als Optionen, die ausgegebenen Texte können konfiguriert werden.

Probleme können entstehen, wenn ein Seitenumbruch in dem ergänzenden Text auftritt. In solchen Fällen kann es passieren, daß der Querverweis ständig zwischen den Seiten hin und her springt und einen Fehler produziert. Mit dem Befehl `\vrefwarning` im Vorspann wird ein solcher Fehler in eine Warnung umgewandelt.

Häufig werden die ersten Querverweise auf Gleitobjekte halbfett gesetzt, damit man die zugehörige Erläuterung schnell findet. In solchen Fällen ist es unerwünscht, daß der gesamte Text halbfett gesetzt wird, etwa **„Abbildung 1 auf Seite 19“.** Dies kann durch Konfiguration der vom `varioref`-Paket benutzten Texte geändert werden:

```
\renewcommand{\reftextcurrent}{\normalfont auf dieser
  Seite}
\renewcommand{\reftextbefore}{\normalfont auf der
  \reftextvario{vorigen}{vorangehenden} Seite}
\renewcommand{\reftextfacebefore}{\normalfont auf der
  \reftextvario{linken}{ge\-gen\-\"uber\-lie\-gen\-den}
    Seite}
\renewcommand{\reftextafter}{\normalfont auf der
  \reftextvario{n\"ach\-sten}{folgenden} Seite}
\renewcommand{\reftextfaceafter}{\normalfont auf der
  \reftextvario{rechten}{ge\-gen\-\"uber\-lie\-gen\-den}
    Seite}
\renewcommand{\reftextfaraway}[1]{\normalfont auf
  Seite~\pageref{#1}}
```

1.1.2 Legendenbildung

Legenden stehen nie in der Grundschriftart, weil sie eine vom Haupttext getrennte Ebene darstellen. Die Schriftart sollte daher kontrastieren, darf aber nicht schwer zu lesen sein. Deshalb bietet sich die Kursive (`\itshape`) an, eventuell auch die Schriftmischung mit der Grotesk (`\sffamily`). Legenden werden flatternd gesetzt (`\raggedright`) und sinngerecht umgebrochen (`\protect\\`).

Die Abstände zum Gleitobjekt müssen stets einheitlich sein. Da LaTeX Legenden stets *unter* dem Gleitobjekt erwartet, stimmen die Abstände nicht, wenn man, wie im Buchdruck üblich, Tabellen mit einer Überschrift versieht. Folgende Kommandos helfen hier weiter:

```
\newcommand{\tablecaption}{%
  \setlength{\abovecaptionskip}{0pt}
  \setlength{\belowcaptionskip}{10pt}
  \caption%
}
\newcommand{\figurecaption}{%
  \setlength{\abovecaptionskip}{10pt}
  \setlength{\belowcaptionskip}{0pt}
  \caption%
}
```

Statt `\caption` wird dann in Tabellen `\tablecaption` benutzt, in Abbildungen `\figurecaption`.

caption2.sty. Dieses Paket erlaubt die einheitliche Formatierung von Legenden. Es verträgt sich mit dem `longtable`-Paket (Abschnitt 1.3.2 auf Seite 13), muß aber nach diesem geladen werden.

```
\usepackage[footnotesize,hang,nooneline]{caption2}
```

erzeugt etwa Legenden in Fußnotengröße, die mit hängendem Einzug versehen sind, auch einzeilige Legenden werden nicht zentriert. Die Schriftart für die Marke und den Legendentext selbst kann etwa mit

```
\renewcommand{\captionfont}{%
  \normalfont\itshape\footnotesize%
}
\renewcommand{\captionlabelfont}{%
  \normalfont\bfseries\footnotesize%
}
```

verändert werden. Das Trennzeichen kann man etwa mit

```
\renewcommand{\captionlabeldelim}{.}
```

anpassen, wenn einem der Doppelpunkt (Voreinstellung) nicht gefällt.

Mit der Option `longtable` wird die Voreinstellung des gleichnamigen Paketes für die Breite der Legenden (4 Zoll) übersteuert und somit eine einheitliche Behandlung aller `\caption`-Befehle erreicht.

1.2 Abbildungen

Bei einspaltigen Texten wird man meist die Abbildungen auf Textbreite skalieren, damit sie sich harmonisch in den Satzspiegel einfügen. Im mehrspaltigen Satz ist darauf zu achten, daß bei der Skalierung von mehreren, nebeneinander gesetzten Bildern sich die Breite der einzelnen Bilder entweder mit der Spaltenbreite deckt oder sich deutlich von ihr unterscheidet. Ebenso sollte die Höhe mehrerer, nebeneinander gesetzter Bilder entweder übereinstimmen oder stark kontrastieren.

1.2.1 Gemachte Betten

graphicx.sty. Dieses Paket gehört zum `graphics`-Bündel und dient der Einbindung von Graphiken. Die Graphikeinbindung ist treiberabhängig, daher muß dem Paket entweder mit einer Option (etwa `dvips`) oder mit einer Datei `graphics.cfg`, die im Eingabepfad für `tex`-Dateien zu finden sein muß, mitgeteilt werden, welcher Treiber verwendet wird. In der Konfigurationsdatei steht etwa ein Eintrag der Form

```
\ExecuteOptions{dvips}
```

Zur Zeit werden u. a. die Optionen `emtex`, `dviwin`, `xdvi` und `dvips` unterstützt, wobei `dvips` die meisten Features bietet.

Der wichtigste Befehl des Paketes ist `\includegraphics` und hat zahlreiche Optionen. Ein typisches Beispiel:

```
\includegraphics[trim=1cm 0cm 2cm
  0.5cm,clip,angle=90,width=\columnwidth]{file}
```

Damit wird die Bounding Box der `eps`-Datei nach innen verschoben, und zwar links um 1 cm, unten um 0 cm, rechts um 2 cm und oben um 0,5 cm. Danach wird die Graphik auf diese neue Bounding Box zurechtgeschnitten, um 90° gedreht (gegen den Uhrzeigersinn) und dann auf Spaltenbreite skaliert. Entsprechend kann man auf etwa 40 % der Texthöhe skalieren, um bei geeignet gewählten Positionierungsparametern mehrere Abbildungen oben auf der Seite zu plazieren. Die Reihenfolge der Optionen ist insbesondere bei der Drehung von Graphiken nicht beliebig!

Für die Graphikdateien kann etwa mit

```
\graphicspath{{eps/}{tiff/}}
```

ein Suchpfad angegeben werden. Standardmäßig werden Graphikdateien in den Verzeichnissen gesucht, in denen auch `tex`-Dateien gesucht werden.

Das `graphicx`-Paket ermöglicht auch das Drehen von Text, etwa in gestürzten Tabellenköpfen (die von rechts lesbar sein müssen):

```
\rotatebox{90}{gestuerzter Eintrag}
```

Gestürzte Tabellenköpfe innerhalb von `landscape`-Umgebungen müssen dagegen um 270° gedreht werden, da sie sonst bei normaler Haltung des Buches auf dem Kopf stünden.

Möchte man grundsätzlich alle Dateien auf die Bounding Box zurechtschneiden und auf Spaltenbreite skalieren, kann man mit

```
\setkeys{Gin}{clip,width=\columnwidth}
```

diese Optionen als Standard festlegen.

1.2.2 Buchstäblich rausgeworfen

psfrag.sty. Dieses Paket erlaubt die Beschriftung von `eps`-Bildern mit LaTeX und verbindet so die einfache Erstellung von Graphiken mit `xfig`, `xmgr` oder CorelDraw mit der Leistungsfähigkeit von LaTeX im Formelsatz. Dazu sind das `graphics`-Bündel sowie ein Postscript-Treiber, etwa `dvips`, erforderlich. Ein typisches Beispiel:

```
\psfrag{marke-in-eps-datei}{$e^{\pi\mathrm{i}}+1=0$}
```

Damit wird eine im Zeichenprogramm zuvor mit der Textfunktion gesetzte Marke durch einen LaTeX-Ausdruck ersetzt. Dieser Befehl muß *vor* dem jeweiligen `\includegraphics`-Befehl stehen, die angegebene Marke *exakt* mit dem Text im Zeichenprogramm übereinstimmen. Eine solche Textersetzung gilt für die gesamte laufende Umgebung, daher kann man direkt nach `\begin{document}` globale Ersetzungen festlegen.

Manche Zeichenprogramme schreiben jeden Buchstaben der Marke einzeln in die Postscript-Datei, so daß die Ersetzung mit `psfrag` nicht mehr funktioniert. In solchen Fällen kann man sich mit Dummy-Texten behelfen, die nur aus einem Buchstaben bestehen.

1.3 Tabellen

1.3.1 Tafeln, wie sie im Buche stehen

Strenge Tabellen, wie man sie in älteren Büchern finden kann, sind meist übersichtlicher als die modernen, mit „Auto Layout“ verunstalteten Varianten: Meist werden statt der Kästchenlösung ausschließlich horizontale Linien verwendet, und die Ausrichtung der Tabelle erfolgt konsequent zur Linksachse. Bei unterschiedlich hohen Tabellenzeilen wirkt ein übergeordnetes Raster beruhigend, bei unklarer Zuordnung der Spaltenüberschriften helfen unterbrochene, horizontale Linien.

booktabs.sty. Dieses Paket stellt Befehle zur Verfügung, mit denen derartige Tafeln auf einfache Weise gesetzt werden können. In der Dokumentation werden weitere Hinweise zum Layout von Tabellen gegeben:

- Keine vertikalen Linien benutzen.
- Keine doppelten Linien benutzen.
- Einheiten stehen im Tabellenkopf.
- Führende Nullen werden ausgeschrieben.
- Keine Unterführungszeichen benutzen, sondern das Feld entweder leer lassen oder den Eintrag wiederholen.

Folgende Befehle stehen u. a. zur Verfügung:

```
\toprule
\midrule
\bottomrule
```

Sie werden zum Setzen der etwas dickeren Kopf- und Fußlinie bzw. der etwas dünneren Halslinie benutzt. Mit

```
\addlinespace
```

schafft man vertikalen Abstand, um unübersichtlich viele Linien in der Tabelle zu vermeiden. Mit

```
\cmidrule(r){1-2}\cmidrule(lr){3-4}\cmidrule(l){5-6}
```

setzt man beispielsweise Linien über die Spalten 1 und 2, 3 und 4 bzw. 5 und 6, die sich nicht berühren. Ohne das Argument in runden Klammern unterbleibt diese „Trimmung". Solche Linien helfen dabei, den Geltungsbereich von kurzen Spaltenüberschriften eindeutig zu klären.

Leider erzeugen die Befehle `\toprule`, `\midrule` und `\bottomrule` bei der Verwendung in `longtable`-Umgebungen Linien über die gesamte Satzbreite. Abhilfe schaffen folgende Befehle,

```
\newcommand{\LTtoprule}[1]{%
  \\
  \cmidrule[\heavyrulewidth]{#1}
}
\newcommand{\LTmidrule}[1]{%
  \cmidrule[\lightrulewidth]{#1}
}
\newcommand{\LTbottomrule}[1]{%
  \cmidrule[\heavyrulewidth]{#1}
}
```

die dann mit etwas Handarbeit (Zählen der Spalten) das Gewünschte leisten, etwa wie folgt:

```
\LTtoprule{1-4}
```

multirow.sty. Dieses Paket stellt den `\multirow`-Befehl zur Verfügung, eine Entsprechung zum `\multicolumn`-Befehl, mit der man mehrere Felder einer Zeile zu einem flatternd gesetzten Feld definierter Breite zusammenfassen kann.

Das erste Argument des `\multirow`-Befehls ist die Zeilenzahl, das zweite die Satzbreite des Feldes. Das dritte, optionale Argument dient zur Feinjustierung der vertikalen Ausrichtung. Der Benutzer muß dafür sorgen, daß die für den `\multirow`-Befehl reservierten Felder tatsächlich *leer* bleiben. Benötigt der Text mehr Zeilen als angegeben, wird der darunter stehende Text überschrieben. Ist zuviel Platz reserviert worden, wird der Text vertikal zentriert, wie im folgenden Beispiel:

Zu	1	Nicht
kurz!	2	zu
	3	kurz!

wurde mit

```
\begin{tabular}{@{}lll}
  \multirow{3}{1cm}{Zu kurz!} & 1 &
  \multirow{3}{1cm}[0.22ex]{Nicht zu kurz!} \\
                              & 2           \\
                              & 3
\end{tabular}
```

erzeugt.

tabularx.sty. Mit diesem Paket kann eine festgelegte Tabellenbreite erreicht werden. Dabei wird *nicht* der Platz zwischen den Spalten vergrößert (wie bei der `tabular*`-Umgebung), sondern die Spalten selbst werden verbreitert. Das folgende Beispiel deklariert eine Tabelle mit zwei linksbündigen Spalten und einer Spalte, deren Breite so festgelegt wird, daß 80 % der Satzbreite erreicht werden:

```
\begin{tabularx}{0.8\textwidth}{llX}
```

1.3.2 Zahlen über Zahlen

longtable.sty. Mit diesem Paket kann man mehrseitige Tabellen automatisch umbrechen lassen. Die `longtable`-Umgebung benutzt den gleichen Zähler wie die `table`-Umgebung, der `\caption`-Befehl erzeugt einen Eintrag in der `lot`-Datei. Horizontale Linien, die mit Seitenumbrüchen zusammenfallen, werden automatisch verdoppelt. Mit dem optionalen Argument können Tabellen linksbündig, zentriert (Voreinstellung) oder rechtsbündig gesetzt werden. Ein Umbruch innerhalb eines `p`-Feldes ist nicht möglich, Fußnoten in `p`-Spalten sind dagegen erlaubt. Möchte man Fußnoten auch im Tabellenkopf verwenden, müssen `\footnotemark` sowie (auf der gleichen Seite im Hauptteil der Tabelle) `\footnotetext` benutzt werden. Ein Tabellenkopf oder -fuß kann automatisch an jedem Seitenbeginn bzw. -ende wiederholt werden. Zur manuellen Steuerung von Umbrüchen sind die normalen LaTeX-Befehle `\nopagebreak` und `\newpage` verwendbar.

Das `longtable`-Paket arbeitet mit `caption2.sty` zusammen und bietet verschiedene Möglichkeiten, Legenden zu erzeugen. Die Überschrift der ersten Tabellenseite erzeugt man etwa mit:

```
\caption[Kurzer Text fuer lot-Datei]{Langer Text}
```

Soll auch auf den weiteren Tabellenseiten eine Überschrift gesetzt werden, kann man hierzu beispielsweise

```
\caption[]{Langer Text, anders als Hauptueberschrift}
```

verwenden. Überschriften werden mit 4 Zoll Satzbreite (Voreinstellung) gesetzt. Den `\label`-Befehl bringt man am besten im Tabellenkopf der ersten Seite unter (vor `\endfirsthead`).

Leider funktioniert das `longtable`-Paket weder mit der `twocolumn`-Option noch mit der `multicols`-Umgebung.

ltxtable.sty. Dieses Paket verbindet die Möglichkeiten von `longtable` und `tabularx`, erlaubt also mehrseitige Tabellen mit automatischem Seitenumbruch und automatischer Anpassung an eine bestimmte Breite.

Die Tabelle steht dabei in einer eigenen Datei und wird etwa wie folgt deklariert:

```
\begin{longtable}{llX}
```

Die Einbindung in das Dokument erfolgt dann etwa mit

```
\LTXtable{0.8\textwidth}{file}
```

array.sty. Dieses Paket erweitert die `tabular`- und die `array`-Umgebung um nützliche Optionen zur Spaltenformatierung. Mit

```
>{\raggedright}p{2.1cm}
```

kann beispielsweise eine flatternd gesetzte Spalte mit definierter Breite deklariert werden, mit

```
>{$}l<{$}
```

erhält man eine linksbündige Spalte für Formeln.

Der Befehl `@{Ausdruck}` *ersetzt* den Abstand zwischen den Spalten durch den angegebenen Ausdruck, der Befehl `!{Ausdruck}` fügt den Ausdruck *zusätzlich* zum Abstand hinzu. Mit

```
p{2cm}m{2cm}b{2cm}
```

werden drei Fließtextspalten erzeugt. Die linke schließt oben bündig mit der laufenden Zeile ab, die mittlere wird vertikal an der laufenden Zeile zentriert, während die dritte unten bündig mit der laufenden Zeile abschließt.

Besonders praktisch ist die Möglichkeit, häufig vorkommende Spaltendeklarationen als eigenen Spaltentyp zu definieren:

```
\newcolumntype{L}{>{$}l<{$}}
```

deklariert einen neuen Spaltentyp `L`, der in Tabellen eine Formelspalte erzeugt. In der `array`-Umgebung würde dagegen auf diese Weise eine Textspalte erzeugt, da das Dollar-Zeichen ein Toggle ist. Auch Kombinationen wie `>{\bfseries}L` sind ohne Schwierigkeiten möglich, ebenso können Argumente benutzt werden:

```
\newcolumntype{P}[1]{>{\raggedright\footnotesize}p{#1}}
```

erzeugt eine flatternd gesetzte Spalte in Fußnotengröße, deren Breite mit dem Argument angegeben werden kann.

dcolumn.sty. Dieses Paket ermöglicht die Ausrichtung von Spalten an Dezimaltrennzeichen. Mit

```
D{,}{.}{2}
```

wird etwa eine Spalte mit zwei Nachkommastellen deklariert, die rechtsbündig im Mathematikmodus gesetzt wird. Dabei wird (z. B. für einen englischen Text) das in der `tex`-Datei verwendete Dezimalkomma für die `dvi`-Datei in einen Dezimalpunkt umgewandelt. Häufig vorkommende Spaltendeklarationen kann man wie beim `array`-Paket abkürzen.

Bei breiten Spaltenüberschriften sind kurze, rechtsbündig ausgerichtete Felder unschön. Besser ist in solchen Fällen, die Zahl mit den meisten Stellen vor dem Komma linksbündig zu setzen und den Rest der Spalte an deren Dezimaltrennzeichen auszurichten. Leider kann man keine linksbündigen Spalten mit Dezimaltrennzeichen deklarieren, sondern lediglich mit beispielsweise

```
D{.}{\cdot}{5.2}
```

alle Zahlen so in der Spalte zentrieren lassen, als hätten sie fünf Vor- und zwei Nachkommastellen. Hier ist also Handarbeit nötig.

2 Pakete – an den Quellen

2.1 Beschaffung

2.1.1 FTP-Server

Die Quelle für Software jeglicher Art zu TEX und LATEX sind die CTAN-Server, in Deutschland ist das

```
ftp://ftp.dante.de/tex-archive/
```

mit zahlreichen Spiegelungen. Ist der Name eines Paketes bekannt, kann man das Suchformular unter der Adresse

```
http://www.dante.de/cgi-bin/ctan-index
```

benutzen, um das Verzeichnis zu finden, in dem es abgelegt ist.

2.1.2 Der LaTeX-Katalog

Sucht man dagegen Informationen darüber, was für Zusatzpakete und Hilfsprogramme es zu TeX und LaTeX überhaupt gibt, hilft der LaTeX-Katalog, eine alphabetisch sortierte Liste, die meist mit Querverweisen zum entsprechenden Verzeichnis versehen ist. Bei einer Stichwortsuche in

```
CTAN: help/Catalogue/catalogue.html
```

findet man meist für das Problem geeignete Pakete.

2.2 Installation

Ein komplettes LaTeX 2_ε-Paket besteht meist aus dem Installationsskript, einer `ins`-Datei, und dem Archiv, einer `dtx`-Datei, die mit Hilfe von `docstrip` bearbeitet wird. Es empfiehlt sich aber immer, die Dokumentation in den jeweiligen Verzeichnissen auf CTAN zu lesen.

2.2.1 Verpackung

Nachdem das Paket auf dem heimischen Rechner vorhanden ist, muß es noch ausgepackt werden. Dies geschieht beispielsweise mit dem folgenden Befehl:

```
tex package.ins
```

Dadurch werden diverse Dateien erzeugt, die noch in die entsprechenden Verzeichnisse verschoben werden müssen.

2.2.2 Dokumentation

Das schönste Paket nützt nicht viel, wenn es nicht dokumentiert ist. Mit dem Befehl

```
latex package.dtx
```

wird die Dokumentation als `dvi`-Datei erzeugt, je nach Paket sind auch noch Durchläufe mit `bibtex` oder `makeindex` notwendig.

2.3 Hilfe

Die schönste Dokumentation nützt nicht viel, wenn sie nicht gelesen wird. Zum Glück lassen sich die meisten Probleme mit der vorhandenen Dokumentation lösen, die in der Regel ausführlich und genau ist. Also: RTFM!

2.3.1 Gedrucktes

Anleitungen

`usrguide.dvi` Hinweise für Umsteiger von LaTeX 2.09, aber auch für Anfänger, da hier das Konzept der Klassen und Pakete erläutert wird. Sollte man kennen.

`fancyhdr.dvi` Hinweise zum Seitenlayout, zur Erstellung von Kopf- und Fußzeilen und zur Plazierung von Gleitobjekten. Sollte man kennen.

`epslatex.ps` Hinweise zur Graphikeinbindung in LaTeX 2_ε. Sollte man kennen.

`modguide.dvi` Hinweise zur Verteilung und Änderung der Bestandteile von LaTeX 2_ε. Für Fortgeschrittene.

`cfgguide.dvi` Hinweise zur Konfiguration von Schriftarten und Trennmustern in LaTeX 2_ε. Für Fortgeschrittene.

`clsguide.dvi` Hinweise für Paket- und Klassenautoren.

`fntguide.dvi` Hinweise zur Auswahl, Installation und Codierung von Schriftarten in LaTeX 2_ε. Für Paket- und Klassenautoren.

Bücher

- KOPKA, H.: *LaTeX*. Addison-Wesley, 1994.
- GOOSSENS, M. et al.: *Der LaTeX-Begleiter*. Addison-Wesley, 1995.
- GOOSSENS, M. et al.: *The LaTeX Graphics Companion*. Addison-Wesley-Longman, 1997.

2.3.2 Elektronisches

FAQ

- `http://www.dante.de/dante/dante-faq.html`

News-Gruppen

- `de.comp.text.tex`
- `comp.text.tex`

Mailing-Listen

- `tex-d-l` (`listserv@vm.gmd.de`)
- `TeX-Euro` (`listserv@urz.uni-heidelberg.de`)
- `emtex-user` (`majordomo@physik.tu-berlin.de`)
- `tetex` (`majordomo@informatik.uni-hannover.de`)