

Die T_EXnische Komödie

dante

Deutschsprachige
Anwendervereinigung T_EX e.V.

18. Jahrgang Heft 3/2006 Juli 2006

3/2006

Impressum

»Die T_EXnische Komödie« ist die Mitgliedszeitschrift von DANTE e.V. Der Bezugspreis ist im Mitgliedsbeitrag enthalten. Namentlich gekennzeichnete Beiträge geben die Meinung der Schreibenden wieder. Reproduktion oder Nutzung der erschienenen Beiträge durch konventionelle, elektronische oder beliebige andere Verfahren ist nur im nicht-kommerziellen Rahmen gestattet. Verwendungen in größerem Umfang bitte zur Information bei DANTE e.V. melden.

Beiträge sollten in Standard-L^AT_EX-Quellcode unter Verwendung der Dokumentenklasse `dtk` erstellt und per E-Mail oder Datenträger an untenstehende Adresse der Redaktion geschickt werden. Sind spezielle Makros, L^AT_EX-Pakete oder Schriften dafür nötig, so müssen auch diese komplett mitgeliefert werden. Außerdem müssen sie auf Anfrage Interessierten zugänglich gemacht werden.

Diese Ausgabe wurde mit Hilfe folgender Programme erstellt: **pdf_eTeXk**, Version 3.141592-1.30.4-2.2 (Web2C 7.5.5). Als Standard-Schriften kamen die Type-1-Fonts Latin-Modern und LuxiMono zum Einsatz.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 2700

Herausgeber: DANTE, Deutschsprachige Anwendervereinigung T_EX e.V.
Postfach 10 18 40
69008 Heidelberg

E-Mail: dante@dante.de
dtkred@dante.de (Redaktion)

Druck: Konrad Triltsch Print und digitale Medien GmbH
Johannes-Gutenberg-Str. 1-3, 97199 Ochsenfurt-Hohe Stadt

Redaktion: Herbert Voß (verantwortlicher Redakteur)

Mitarbeit : Rudolf Herrmann Gerd Neugebauer Andreas Scherer
Lutz Ihlenburg Rolf Niepraschk
Gert Ingold Volker RW Schaa

Redaktionsschluss für Heft 4/2006: 15. Oktober 2006

ISSN 1434-5897

Die T_EXnische Komödie 3/2006

Editorial

Liebe Leserinnen und Leser,

diese Ausgabe unserer Vereinszeitschrift entstand unter Hochdruck, wobei hier diesmal nur das aktuelle Klima gemeint ist; die Lüfter der PCs tragen nicht unbedingt zur Frischluftzufuhr bei diesen europaweiten hochsommerlichen Temperaturen bei ...

In diesem Heft finden Sie einen weiteren Zwischenbericht zum $\epsilon\chi$ TeX-Projekt, welches mittlerweile schon recht weit vorangeschritten erscheint. Ans »Eingemachte« geht es im Artikel über Zähler und ans Praktische in den Artikeln zu Randbemerkungen und einer grafischen Oberfläche für das Paket `dinbrief`. Typografische Konventionen sind gerade im Zusammenhang mit TeX immer wieder auf der Tagesordnung, denn allzu schnell gerät in Vergessenheit, was einmal allgemeiner Konsens war.

Die Rubrik der neuen Pakete ist diesmal etwas länger geraten, da diese beim letzten Mal aus gegebenem Anlass ausfallen musste.

Ich wünsche Ihnen, dass Sie wieder Interessantes und Neues in dieser Ausgabe finden und verbleibe

mit TeXnischen Grüßen,

Ihr Herbert Voß

Hinter der Bühne

Vereinsinternes

Grußwort

Klaus Höppner, Volker RW Schaa

Liebe Mitglieder,

dieses Grußwort erreicht Sie wahrscheinlich entweder kurz vor oder nach Ihrem Sommerurlaub. Den einen wünschen wir, dass sie sich gut erholt haben, den anderen, dass sie es noch tun mögen.

In dieser $\text{\TeX}$ nischen Komödie finden Sie die Einladung zur diesjährigen Herbsttagung. Nachdem die letzten Tagungen eher nördlich stattfanden, gehen wir diesmal zur Fachhochschule nach Rosenheim, wo sich Prof. Georg Lachenmayr zur Organisation der Tagung bereit erklärt hat. Aufgrund der guten Verkehrsanbindung Richtung Salzburg ist diese Veranstaltung natürlich auch für österreichische Mitglieder interessant.

Seit der DANTE 2006 in Berlin gab es zwei wichtige Veranstaltungen, den Linuxtag in Wiesbaden und die Euro $\text{\TeX}$ in Ungarn.

Die Aktiven des Stands von DANTE e.V. auf dem Linuxtag waren hervorragend eingespielt, war es doch schon die zweite Teilnahme. Karlheinz Geyer hatte wieder den Hauptanteil der Organisation geleistet, und mit Karin Dornacher, Klaus Höppner, Jens Langecker, Martin Wilhelm Leidig, Frank Mohr, Heiko Oberdiek, Meike Schmedt, Martin Schröder, Jan Steffan und Ulrik Vieth waren viele Aktive des letzten Jahres wieder vertreten. Ihnen allen gilt unser herzlicher Dank. Der Linuxtag stand unter dem Vorzeichen vieler Veränderungen: ein neuer Veranstaltungsort, ein neuer Termin und eine neue Organisationsleitung. Wie Sie vielleicht der Presse entnommen haben, ist die Besucherzahl wiederum um etwa 25 % gesunken. Es war zwar etwas ruhiger, das Publikum war aber insbesondere an den Ständen der freien Projekte sehr interessiert, sodass wir durchaus zufrieden waren. Bei kommerziellen Ausstellern war aber eine deutliche Enttäuschung über das zurückhaltende Publikum

spürbar. Insofern erscheint die Zukunft des Linuxtages ungewisser als das Interesse von DANTE e.V. an einer erneuten Teilnahme.

In der Zeit vom 4. bis 8. Juli fand die 16. EuroT_EX-Tagung in Debrecen, Ungarn statt. Debrecen beherbergt die größte Universität Ungarns, bei der wir mit dieser Tagung zu Gast waren. Da ein ausführlicher Artikel über die Tagung für die nächste Ausgabe von »Die T_EXnische Komödie« geplant ist, soll hier nur ein knapper Bericht gegeben werden. Die EuroT_EX glänzte als »All Inclusive«-Tagung mit hervorragender Organisation, kurzen Wegen zwischen Vortragssälen, Campushotel und Verpflegungsstätten und konnte mit einem gut gestalteten Programm die Erwartungen noch übertreffen. Leider waren nur wenige T_EXies der Einladung gefolgt und so wurden 33 Teilnehmer intensiv durch das Organisationskomitee betreut. Über die T_EXnischen Highlights, den Puszta-Ausflug und alle besonderen Vorkommnisse lesen Sie in der nächsten Ausgabe von »Die T_EXnische Komödie«...

Viele positive Rückmeldungen haben wir zum Tagungsband der EuroT_EX-2005 erhalten. Der lange Zeitraum seit der Konferenz mag Ihnen ein Hinweis für den Aufwand des Sammelns der Beiträge und des Zusammenfügens dieser zu einem ansprechenden Ganzen sein. Wer zusätzliche Exemplare des Tagungsbandes bestellen möchte, kann dies gerne für 10 Euro (inkl. Versand) über die Geschäftsstelle tun. Auch zusätzliche Exemplare der T_EX-Aufkleber, die der letzten Lieferung beilagen, können dort bestellt werden (kleiner Aufkleber jeweils 20 Cent, großer Aufkleber je 50 Cent, zzgl. 55 Cent Porto).

Wir möchten Sie noch einmal auf die Möglichkeit einer vergünstigten Mitgliedschaft in der TUG, der internationalen T_EX Users Group, aufmerksam machen. Wenn Sie für das Jahr 2006 zusätzlich Mitglied der TUG werden möchten (Zusatzbetrag 45,50 Euro, ermäßigt 25 Euro), wenden Sie sich bitte an die Geschäftsstelle von DANTE e.V. Kombinierte Mitglieder gelten als vollwertige Mitglieder der TUG, erhalten aber nicht die jährliche DVD-/CD-Sendung und gemeinsame Veröffentlichungen, die bereits Teil der DANTE-Mitgliedschaft sind. Aus diesem Grund erhalten kombinierte Mitglieder dieses Jahr nur zwei Ausgaben der Zeitschrift TUGboat, da der Tagungsband der EuroT_EX 2005 eine TUGboat-Ausgabe ersetzt.

Mit freundlichem Gruß,

Klaus Höppner	Volker RW Schaa
Vorsitzender	Stellv. Vorsitzender

Einladung und »Call for Papers« zur Herbsttagung von DANTE e.V.

Klaus Höppner, Georg Lachenmayr

Liebe Mitglieder von DANTE e.V.,

die nächste TeX-Tagung von DANTE e.V. findet am 14. und 15. September 2006 an der Fachhochschule Rosenheim statt.

Am Donnerstag sind Tutorien und Vorträge geplant, am Freitag wird die 35. Mitgliederversammlung von DANTE e.V. stattfinden.

Die Tagesordnung der Mitgliederversammlung am 15. 9. 2006 um 9.00 Uhr an der

Fachhochschule Rosenheim
Hörsaal B 023
Hochschulstraße 1
83024 Rosenheim

lautet:

1. Begrüßung und Tagesordnung
2. Bericht des Vorstands
3. Bereitstellung neuer Mittel für den Projektfonds
4. Verschiedenes

Ihre Stimmunterlagen erhalten Sie direkt vor Ort, um vorherige Anmeldung wird gebeten. Eine Übertragung des Stimmrechts ist im Rahmen des § 13 (4) der Vereinssatzung möglich. Wie üblich sind auch Nichtmitglieder als Gäste willkommen.

Falls Sie ein Tutorium oder einen Vortrag anbieten wollen, werden Sie gebeten, dies mit dem Anmeldeformular unter <http://www.dante.de/dante/events/mv35/> oder per E-Mail an mv35@dante.de bei den Organisatoren möglichst bis 15. August 2006 anzumelden.

Zu einem Vortrag oder Tutorium ist ein Abstract einzureichen. Richtlinien dafür sind auf der oben genannten WWW-Seite zu finden oder bei den Organisatoren zu erfragen.

Firmen und Institutionen, die ihre Produkte präsentieren bzw. die Tagung finanziell unterstützen wollen, werden gebeten, sich frühzeitig an dieselben Adressen zu wenden.

Die Homepage der Tagung findet sich unter <http://www.dante.de/dante/events/mv35/>

Mit Fragen, Wünschen und Anregungen wenden Sie sich bitte an

DANTE e.V.

Stichwort: Mitgliederversammlung von DANTE e.V.

Postfach 10 18 40

69008 Heidelberg

E-Mail: mv35@dante.de

Mit freundlichen Grüßen

Klaus Höppner (DANTE e.V.)

Georg Lachenmayr (FH Rosenheim)

Bretter, die die Welt bedeuten

Was hinten rauskommt zählt: Counter Aliasing in $\text{\LaTeX}$

Ulrich Schwarz¹

Für manche Einsatzzwecke ist es interessant, mehrere Zähler zu haben, die getrennte Namen und `\the...`-Darstellungen haben, aber sich einen Zählerwert teilen. Wir studieren dies am Beispiel von `hyperref` und `theorem`-Umgebungen.

Nomen nescio

Es ist sicherlich keine weit hergeholte Anwendung: man arbeitet an einem größeren Werk, in dem mehrere theoremartige Umgebungen sich eine Zählung teilen, wie in vereinfachter Form in folgendem Listing dargestellt:

```
1 \documentclass{article}
2 \usepackage{hyperref}
3 \newtheorem{satz}{Satz}
4 \newcommand\satzname{Satz}% Bezeichnung für autoref.
5 \newtheorem{hauptsatz}[satz]{Hauptsatz}
6 \newcommand\hauptsatzname{Hauptsatz}% Bezeichnung für autoref.
7 \begin{document}
8   \begin{hauptsatz}[Rekursive Verweise]\label{1}
9     Dies ist \autoref{1}.
10  \end{hauptsatz}
11 \end{document}
```

¹Der Autor dankt Heiko Oberdiek für fruchtbare Diskussion darüber, wer wann was wo hinzufügen soll, die zu dieser Version mit umfassenderer Funktionsfähigkeit geführt hat.

Hauptsatz 1 (Rekursive Verweise) *Dies ist Satz 1.*

Abbildung 1: Synchron laufende Zähler und `\autoref`.

Es wird allerdings verwundern, dass das Ergebnis nicht wie gewünscht ausfällt, sondern unser Hauptsatz von `\autoref` zu einem Satz degradiert wird, wie in Abb. 1 zu sehen.²

Nach einigem Gestöber in den Quelltexten [5] lässt sich dies folgendermaßen erklären: `hyperref` ist zur Laufzeit darauf angewiesen, `\refstepcounter` abzufangen, um zusätzliche Informationen zu erhalten. Insbesondere richtet sich das `\...name`-Makro bei der Ausgabe des Verweises nach dem Namen des Zählers, der erhöht wurde. In unserem Falle ist dies nun leider der Zähler **satz**, den wir explizit im `\newtheorem`-Befehl verlangt hatten. Wir möchten das Problem umgehen, aber tunlichst vermeiden, `\newtheorem` und Konsorten umzudefinieren.³

Tarnen und täuschen

Ein Lösungsansatz bietet sich an: wir geben `\newtheorem` zwar weiter einen ausdrücklichen Zähler vor, schieben aber in Wirklichkeit ein präpariertes Konstrukt unter: es soll immer den Zählerstand von **satz** widerspiegeln, aber einen anderen Namen haben, damit ein anderes `\...name`-Makro verwendet wird. Hierzu ist es hilfreich, sich ins Gedächtnis zu rufen, was für L^AT_EX alles zu einem Zähler **satz** gehört:

Primitives Zählerregister Um den Zahlenwert selbst zu speichern, reserviert L^AT_EX eines der T_EXschen Zählerregister `\count<num>` und macht es unter dem Namen `\c@satz` zugänglich.

Darstellungsmakro Die bevorzugte gedruckte Darstellung des Zählerwertes soll vom Makro `\thesatz` geliefert werden.

Präfixstring Das Makro `\p@satz` wird durch `\refstepcounter` vor »die Nummer« `\thesatz` gesetzt. Dieses Makro erzeugt standardmäßig keine Aus-

²Der misstrauische Leser möge sich überzeugen, dass die in der Abbildung gezeigte manuelle Nachbildung dem Ergebnis eines L^AT_EX-Laufs entspricht.

³Man bedenke, dass hier auch Umdefinitionen für `theorem`, `amsthm`, `ntheorem`, `shadethm` etc. fällig werden. Wehe dem, der sich sonst später entscheidet, das Paket zu wechseln!

gabe, kann aber direkt oder mit dem `\labelformat`-Befehl des Pakets `varioref` [2] gesetzt werden.

Rücksetzliste Wenn ein Zähler erhöht wird, sollen unter Umständen andere zurückgesetzt werden. Alle diese zurückzusetzenden Zähler sind im Makro `\cl@satz` in der Form `\@elt{ctr1}\@elt{ctr2}\@elt{ctr3}` gespeichert, so dass durch geeignetes lokales Definieren des Befehls `\@elt` verschiedene Aktionen durchgeführt werden können.

Eintrag in \cl@ckpt Diese Rücksetzliste wird für das Abspeichern der Zählerstände benutzt, damit bei Verwendung von `\includeonly` die Zähler auf die richtigen Werte (nämlich die am Ende des Vorkapitels) initialisiert werden können.

Mindestens zwei weitere Makros kommen durch `hyperref` hinzu:

saubere Darstellung Zusätzlich zu `\thesatz` ist es ratsam, `\theHsatz` zu definieren. Diese Darstellung des Wertes wird von `hyperref` intern verwendet und landet unter anderem in der `aux`-Datei, weshalb man auf allzu exotische Darstellungen verzichten sollte. Sie muss allerdings für das gesamte Dokument eindeutig sein.

autoref-Namen Schlussendlich gibt es noch die Makros `\satzname` und `\satzautorefname`, mit denen sich der Text von `\autoref` steuern lässt. (Hierbei wird `\satzautorefname` bevorzugt benutzt, weil `\...name` teilweise auch anderweitig, zum Beispiel von `babel`, benutzt wird. Wir benutzen hier der Übersichtlichkeit halber aber die kürzere Form.)

Es sollte nun reichen, diese Makros zu kopieren, um unseren gewünschten Zähleralias zu bekommen:

```

1 %% make #1 a counter that uses counter #2's count register.
2 \newcommand*{@counteralias[2]{\begingroup%
3   \def@@gletover##1##2{%
4     \expandafter\global
5     \expandafter\let\csname ##1\expandafter\endcsname
6     \csname ##2\endcsname
7   }%
8   \@ifundefined{c@#2}{\@nocounterr{#2}}{%
9     \expandafter\@ifdefinable\csname c@#1\endcsname{%
10      \@@gletover{c@#1}{c@#2}%
11      \@@gletover{the#1}{the#2}%
12      \@@gletover{theH#1}{theH#2}%

```

```

13      \@@gletover{p@#1}{p@#2}%
14      \expandafter\gdef\csname c1@#1\expandafter\endcsname
15      \expandafter{\csname c1@#2\endcsname}%
16    }%
17  }%
18 \endgroup}

```

Wir liefern hier keine Vorbelegung der `\...name`-Makros, da diese wohl sowieso vom Benutzer geändert werden. (Ändert man `\newtheorem` im Sinne unserer Ausführungen ab, so kann man natürlich direkt den vom Benutzer gegebenen Namen zur Initialisierung benutzen.) Des weiteren definieren wir immer ein `\theH...`-Makro, auch wenn `hyperref` nicht geladen ist – in diesem Falle wird es `\relax`. Einen weiteren Eintrag in `\c1@ckpt` brauchen wir nicht: das Setzen des Originalzählers wird auch den Wert des Alias-Zählers richtig setzen.

Schließlich ist noch zu bemerken, dass die Rücksetzliste hier eine Ausnahme stellt: wir kopieren nicht die Definition der Originalliste, sondern lassen die Aliasliste die Originalliste aufrufen. Zweck dieses Umwegs soll sein, dass Änderungen an der Originalliste sich automatisch auf den Alias auswirken. Aber dafür müssen wir jetzt »die Ärmel hochkrempeln«.

Alles auf Anfang!

Tragen wir die nötigen Definitionen aus dem L^AT_EX-Kernel [4] zusammen, so ist der interne Befehl `\addtoreset` wie folgt realisiert:

```

1 \def\@cons#1#2{\begingroup
2   \let\@elt\relax
3   \xdef#1{#1\@elt #2}%
4 \endgroup}
5 \def\@addtoreset#1#2{%
6   \expandafter\@cons\csname c1@#2\endcsname {{#1}}%
7 }

```

Das Erweitern funktioniert also über den bereits erwähnten `\@elt`-Mechanismus und vollständige Expansion. So erfährt zwar der Alias von Änderungen am Original, aber sobald ein Zähler zur Aliasliste hinzugefügt wird, geht diese Verbindung durch das `\xdef` verloren. Änderungen am Alias erfährt das Original gleich gar nicht. Wir werden uns also die Mühe machen

müssen, `\@addtoreset` entsprechend anzupassen. Insbesondere müssen wir »echte« Rücksetzlisten von Zeigern auf andere Rücksetzlisten unterscheiden. Konkret nutzen wir aus, dass solch eine echte Liste leer ist oder mit `\@elt` beginnt.⁴ Genauer: wir behandeln *genau die* Listen als echt, die leer sind oder mit `\@elt` beginnen.

```

19 \newcommand*\aliasctr@follow[1]{%
20   \expandafter\aliasctr@f@llow
21   \csname cl@#1\endcsname\@elt\@nil{\csname cl@#1\endcsname}%
22 }
23 \def\aliasctr@f@llow#1#2\@nil#3{%
24   \ifx#1\@elt
25     % Liste beginnt mit \@elt oder ist leer -> echte Liste
26     \noexpand #3%
27   \else
28     % #1 ist selbst ein \@cl-Makro -> rekursiv suchen
29     \expandafter\aliasctr@f@llow#1\@elt\@nil{#1}%
30   \fi
31 }

```

Hierbei ist zu beachten, dass immer mindestens ein rekursiver Aufruf stattfindet, da `\aliasctr@follow` das *Token* `\cl@#1` übergibt und nicht seine Expansion, die Vorbelegung des dritten Parameters ist also beliebig.

Mit diesen Vorbereitungen können wir jetzt folgende Definition wagen, die sich im Wesentlichen durch das Auffinden der Originalliste unterscheidet:

```

32 \renewcommand*\@addtoreset[2]{\bgroup
33   \edef\aliasctr@@truelist{\aliasctr@follow{#2}}%
34   \expandafter\@cons\aliasctr@@truelist{{#1}}%
35 \egroup}

```

Zurück, marsch marsch

Auf analoge Weise können wir nun auch einen Ersatz für `\@removefromreset` aus David Carlisles `remreset`-Paket [1] definieren:

```

36 %% Slight modification of remreset.sty.
37 \newcommand*\@removefromreset[2]{\bgroup

```

⁴ Auch andere Konstruktionen, beispielsweise das Speichern des Originalcounternamens in einem weiteren Makro, sind möglich, siehe [3].

```

38 \edef\aliasctr@@truelist{\aliasctr@follow{#2}}%
39 \expandafter\let\csname c@#1\endcsname\@removefromreset
40 \def\@elt##1{%
41   \expandafter\ifx\csname c@#1\endcsname\@removefromreset
42   \else
43     \noexpand\@elt{##1}%
44   \fi}%
45 \expandafter\xdef\aliasctr@@truelist{%
46   \aliasctr@@truelist}
47 \egroup}

```

Eine letzte Hürde

Kehren wir nun mit unseren Definitionen zu unserem Beispiel zurück, so möchten wir jetzt folgendes schreiben:

```

1 \newtheorem{satz}{Satz}
2 \newcommand\satzname{Satz}% Bezeichnung für autoref.
3 \makeatletter
4 \@counteralias{hauptsatz}{satz}
5 \makeatother
6 \newtheorem{hauptsatz}[hauptsatz]{Hauptsatz}
7 \newcommand\hauptsatzname{Hauptsatz}% Bezeichnung für autoref.

```

Hier laufen wir jetzt in die letzte Falle: zwar definiert `\newtheorem` nun keinen eigenen Zähler mehr, allerdings wird immer noch `\thehauptsatz` definiert, und zwar unter Rückgriff auf das Ausgabemakro des übergebenen Zählers, hier also bereits `\thehauptsatz`, was bei der Benutzung zu einer Endlosschleife führt. Wir müssen also diese Definition (die sich im Wesentlichen tatsächlich auf `\def\thehauptsatz{\thehauptsatz}` beschränkt) wieder rückgängig machen:

```

8 \let\thehauptsatz\thesatz

```

Damit läuft unser Beispiel endlich wieder durch und liefert die Ausgabe aus Abbildung 2.

Zusammenfassung

Wir haben einen Ansatz gesehen, der es erlaubt, verschiedene Theoremtypen zu unterscheiden, auch wenn diese sich einen Zähler teilen. Dabei haben wir

Hauptsatz 1 (Rekursive Verweise) *Dies ist Hauptsatz 1.*

Abbildung 2: \@counteralias und \autoref.

keine count-Register belegt, die wir nicht benutzen. Diese Vorgehensweise ist natürlich weder auf Theoreme beschränkt, auch wenn diese ein prominentes Beispiel sind, noch auf die Zusammenarbeit mit `hyperref` oder `autoref`. Zwei Pakete [3, 6] setzen den hier vorgestellten Ansatz um; für `aliasctr` [6] ist geplant, in näherer Zukunft die `\newtheorem`-Befehle automatisch entsprechend zu erweitern und bei der Gelegenheit weitere Funktionalität (beispielsweise Theoremlisten ähnlich denen von `ntheorem`) zu bieten.

Literatur

- [1] David Carlisle: *The `remreset` package*; Sept. 1997; CTAN://macros/latex/contrib/carlisle/remreset.sty.
- [2] Frank Mittelbach: *The `varioref` package*; Febr. 2004; CTAN://macros/latex/required/tools/varioref.dtx.
- [3] Heiko Oberdiek: *The `aliascnt` package*; 2006; CTAN://macros/latex/contrib/oberdiek/aliascnt.dtx, to appear.
- [4] The L^AT_EX3 Project: *The L^AT_EX 2_ε Sources*; Dez. 2003.
- [5] Sebastian Rahtz und Heiko Oberdiek: *The `hyperref` package*; Nov. 2003; CTAN://macros/latex/contrib/hyperref/.
- [6] Ulrich M. Schwarz: *The `aliasctr` package*; 2006; zur Zeit <http://www.informatik.uni-kiel.de/~ums/aliasctr.tar.gz>.

Marginalien, da wo man sie haben will!

Michael Niedermair, Markus Kohm

In Mailinglisten und Newsgroups liest man immer von dem Problem, dass Marginalien nicht an der Stelle stehen, an der man sie haben will oder dass diese die Fußnoten so beeinflussen, dass sie auf einer anderen Seite landen oder dass Marginalien an Stellen benötigt werden, an denen sie nicht möglich sind. Das Paket *marginnote* schafft hier Abhilfe.

Einleitung

Marginalien, auch Randnotizen genannt, werden bei L^AT_EX mit dem Befehl `\marginpar{}` gesetzt. Leider werden diese aber durch weitere Marginalien, durch den Umbruch oder andere gleitende Objekte beeinflusst. Dadurch bekommt man manchmal Meldungen wie »*Marginpar on page . . . moved*«. Gerade im Zusammenhang mit Fußnoten kann es dazu kommen, dass die Fußnote auf der nächsten Seite gesetzt wird, was nicht erwünscht ist. Hinzu kommt, dass `\marginpar{}` die Randnotiz als gleitendes Material behandelt. Dadurch ist es nicht möglich, Marginalien innerhalb anderer Gleitumgebungen zu verwenden. Während es bei Abbildungen noch relativ leicht ist, mit geeigneten Boxen und `\hspace`-Anweisungen dennoch in den Rand zu gelangen, wird dies spätestens bei Verwendung des **framed**-Paketes zu einem Problem.

Das Paket »marginnote«

Markus Kohm hat auf der Mailingliste *T_EX-D-L* ein Beispielmakro veröffentlicht, mit dem die oben beschriebenen Probleme gelöst und die Marginalien dort gesetzt werden, wo sie sein sollen. Durch ein paar Nachfragen ist daraus das Paket *marginnote* entstanden, welches den Befehl `\marginnote` zur Verfügung stellt.

```
\marginnote[<links>]{<rechts>}[<vertikale Verschiebung>]
```

Die ersten beiden Argumente entsprechen denen von `\marginpar`. Mit dem weiteren optionalen Parameter lässt sich die Marginalie vertikal verschieben. Die Voreinstellung ist dabei 0. Durch das Umdefinieren des Befehls

Javascriptcode für Version 7 wird mit einem Icon versehen. Hier kommt ein Beispielcode.

normale Randnotiz

```
activeDocument = documents[0];
if (activeDocument.width.value >
    activeDocument.height.value)
alert("Format: Querformat");
```



Abbildung 1: Ergebnis von Listing 1

`\marginfont` kann man einen anderen Font oder auch Farbe für die Marginalie verwenden.

Während `\marginpar` die Marginalien im Blocksatz setzt, werden sie in den Voreinstellungen bei `\marginnote` im linken Rand rechtsbündig und im rechten Rand linksbündig gesetzt. Durch das Umdefinieren der Befehle `\raggedleftmarginnote` und `\raggedrightmarginnote` kann man die Ausrichtung der Marginalie im linken bzw. rechten Rand ändern.

Im nachfolgenden Beispiel 1 werden zwei Marginalien mit dem Befehl `\marginpar` gesetzt. Die erste enthält einen normalen Text, die zweite ein Icon, welches direkt neben dem Listing (an der Oberkante ausgerichtet) platziert werden soll.

Listing 1: Beispiel mit *marginpar*

```
22 \def\icon{\raisebox{-3em}{%
23   \includegraphics[height=2em]{ps_7}}}
24 \noindent%
25 Javascriptcode für Version 7 wird mit einem
26 Icon\marginpar{normale Randnotiz} versehen.
27 Hier kommt ein Beispielcode.
28 \marginnote{\icon}%
29 \begin{lstlisting}
30 activeDocument = documents[0];
31 if (activeDocument.width.value >
32     activeDocument.height.value)
33 alert("Format: Querformat");
34 \end{lstlisting}
```

Dabei verursacht die erste Marginalie, dass die zweite nach unten verschoben wird und somit das Listing mit dem Icon nicht mehr bündig ist (siehe Abbildung 1).

Javascriptcode für Version 7 wird mit einem Icon versehen. Hier kommt ein Beispielcode.

```
activeDocument = documents[0];
if (activeDocument.width.value >
    activeDocument.height.value)
    alert("Format: Querformat");
```

normale Randnotiz



Abbildung 2: Icon korrekt gesetzt.

Ersetzt man dagegen `\marginpar{\icon}` durch `\marginnote{\icon}`, so wird das Icon richtig platziert (siehe Abbildung 2).

Doch das Paket bietet weitere Einsatzmöglichkeiten. Will man etwa neben einer Fußnote¹ eine Marginalie unterbringen, so scheitert `\marginpar`. Mit `\marginnote` ist dies jedoch möglich, wie die Beispielfußnote zeigt.

Auch innerhalb von Gleitumgebungen ist die Verwendung von `\marginnote` im Gegensatz zu `\marginpar` möglich. Dies ist von besonderer Bedeutung, wenn das `framed`-Paket verwendet wird.

Allerdings kann auch das `marginnote`-Paket im Zusammenhang mit `framed` zunächst nicht alle Probleme lösen. Die folgende `framed`-Umgebung zeigt ein solches Problem.

Hier ist zu beobachten, dass die horizontale Platzierung des Fragezeichens in dieser `framed`-Umgebung nicht korrekt ist. Dies liegt an der Arbeitsweise von `framed`.

?

Das Paket `framed` setzt die Länge `\textwidth` auf die tatsächliche Breite des Textes innerhalb der jeweiligen Umgebung. In den meisten Fällen ist dies korrekt. `marginnote` jedoch verwendet diese Länge, um den rechten Rand zu erreichen. Bei der `framed`-Umgebung müsste dazu aber noch der Abstand zur Kastenlinie überwunden werden.

Ganz ähnlich verhält es sich mit dem linken Rand. Auch dieser ist um `\FrameSep` weiter vom normalen Textanfang entfernt.

¹Dies ist ein Beispiel für eine Fußnote, die mit einem Icon als Marginalie versehen wurde.



!

Im Beispiel des Ausrufezeichens wurde dies durch Einfügen einer `\hspace*`-Anweisung in den beiden Marginalen-Argumenten von `\marginnote` erreicht:

```
1 \marginnote[\Huge!\hspace*{\FrameSep}]
2      {\hspace*{\FrameSep}\Huge!}
```

Diese Lösung ist jedoch nicht sonderlich komfortabel, da im doppelseitigen Layout unbedingt beide Argumente verwendet werden müssen und die `\hspace*`-Anweisungen bei jeder `\marginnote`-Anweisung einzufügen sind.

Doch auch für solche Probleme bietet das Paket `marginnote` eine Lösung. Kennt man den horizontalen Versatz, der innerhalb einer Umgebung notwendig ist, so kann man die Makros `\marginnoterightadjust` (für den rechten Rand) und `\marginnoteleftadjust` (für den linken Rand) entsprechend definieren.

!

In diesem Beispiel wurde nun der notwendige horizontale Versatz einfach erreicht, indem nach `\begin{framed}` die Anweisungen

```
1 \let\marginnoterightadjust\FrameSep
2 \let\marginnoteleftadjust\FrameSep
```

eingefügt wurden. Die Marginalie konnte dadurch einfach mit:

```
1 \marginnote{\Huge!}
```

gesetzt werden.

Im Falle der realen Anwendung ist es natürlich sinnvoller, diese Anweisung in die Definition der `framed`-Umgebung aufzunehmen. Dies ist mit

```
1 \begingroup
2   \makeatletter
3   \g@addto@macro\framed{%
4     \let\marginnoterightadjust\FrameSep
5     \let\marginnoteleftadjust\FrameSep
6   }
7 \endgroup
```

auch nachträglich einfach möglich.

Trotzdem hat auch das Paket `marginnote` seine Grenzen. So ist es beispielsweise innerhalb mathematischer Umgebungen nur eingeschränkt verwendbar. Während die Verwendung in einer einfachen `displaymath`-Umgebung eventuell noch erfolgreich ist, dürfte die Anweisung in den meisten Umgebungen des `amsmath`-Paketes versagen.

Gegenüber der einfachen `\marginpar`-Anweisung hat das Paket einen weiteren Nachteil. Beim Setzen der Marginalie ist im doppelseitigen Satz noch nicht bekannt, welcher Rand der richtige ist. Dies ist erst bekannt, wenn die Seite tatsächlich ausgegeben wird. Das Paket verwendet einen `\label`-verwandten Mechanismus, um diese Information in den nächsten `LATEX`-Lauf zu retten. So lange am Ende des `LATEX`-Laufes die bekannte Meldung »*Labels(s) may have changed. Rerun to get cross-references right*« erscheint, ist davon auszugehen, dass einzelne Marginalien noch falsch platziert sind.

Zusammenfassung

Mit dem Paket `marginnote` lassen sich Marginalien da setzen, wo sie hin gehören. Darüber hinaus ermöglicht das Paket auch in vielen Fällen das Setzen von Marginalien, in denen `\marginpar` komplett versagt.

typokurz – Einige wichtige typografische Regeln

Christoph Bier

Dieses Dokument soll einen *schnellen* Überblick über ein paar aus meiner Sicht wichtige und gleichzeitig leicht einzuhaltende (mikro)typografische Regeln geben, die die Leserlichkeit eines Textes erhöhen. Daher spare ich mir lange Erklärungen und Diskussionen im Text. Das Original und aktuelle Versionen einschließlich kommentierter Präambel finden sich unter <http://www.zvisionwelt.de/downloads.html>. Grundlage für diesen Artikel war Version 1.4 vom 14. Juni 2005.

Einleitung

Mit den Regeln in **typokurz** sind die typografischen Fehler abgedeckt, die mir beim Lesen verschiedenster Texte am häufigsten aufgefallen sind. Dazu gehören Briefe, Bewerbungen, Vorlesungsskripte, Referate, Hausarbeiten, Diplomarbeiten, aber auch wissenschaftliche Artikel für Journale, die noch nicht vom Verlag/Satzbüro/Lektor korrigiert wurden.

Wer diese Regeln beachtet, erleichtert dem Leser das Erfassen des Inhalts. Mir wird an dieser Stelle immer wieder gerne entgegnet, dass es doch wohl nur auf den Inhalt und nicht die Form ankäme. Dies ist einerseits ein Irrtum und andererseits enthalten beispielsweise Satzzeichen selbst inhaltliche Informationen. Es macht nun mal einen Unterschied, ob an einer Stelle im Text ein Gedankenstrich oder ein Bindestrich steht oder ein Ausrufezeichen oder ein Fragezeichen. Form und Inhalt gehören zusammen; ein roter Klecks ist etwas anderes als ein rotes Herz. Außerdem achten doch die allermeisten sehr genau auf eine korrekte Orthografie. Wer diesbezüglich auf den Duden schwört, findet auch dort Regelungen und Hinweise, wie Anführungszeichen, Gedankenstriche usw. zu setzen sind. Natürlich geht die Typografie weit über diese Dinge hinaus. Doch diese in **typokurz** zu behandeln, würde den Rahmen sprengen. Es geht um für jedermann (und für jede Textverarbeitung!) leicht umsetzbare Regeln. Gerade Letzteres ist auch der Grund, warum **typokurz** systemunabhängig gehalten ist. Ein Artikel, wie man die Regeln konkret mit L^AT_EX umsetzt, ist geplant. Allerdings sei an dieser Stelle auch auf die Artikel

von Marion Neubauer hingewiesen [4, 5], die das Thema ausführlicher und speziell für L^AT_EX behandeln. Ich hoffe, dass sich auch Menschen finden, die die Umsetzung von typokurz für Textverarbeitungen beschreiben.

Auszeichnung/Hervorhebung von Text

Es gibt verschiedene Möglichkeiten, einzelne Wörter oder Passagen in einem Text auszuzeichnen, die hier genannt und kurz erläutert werden.

Kursive: eigene Schriftform; *integrierte* Auszeichnung, die erst auffällt, wenn man an die entsprechende Stelle kommt; im Normalfall für Auszeichnungen im Text am besten geeignet.

Schräge: Schrägstellung der Aufrechten; keine eigene Schriftform; »Eine [...] elektronisch schräggestellte Schrift ist keine Kursive sondern ein Mißverständnis.« [8]. (Serifenlose Schriften haben häufig keine Kursive.)

Fette: normalerweise in Textabschnitten zu vermeiden, viel zu aufdringlich (*aktive* Auszeichnung), zieht direkt die Aufmerksamkeit auf sich; für Überschriften, Bezeichnungen von Tabellen und Abbildungen, für Teile von bestimmten Aufzählungen und Verzeichnissen und Tabellenköpfe geeignet; gelegentlich findet man sie auch für Literaturverweise.

Unterstreichung: unbedingt zu vermeiden; Überbleibsel aus dem Schreibmaschinenzeitalter, als es nur eine Schriftform – die Aufrechte – auf der Schreibmaschine gab.

VERSALIEN: (= Großbuchstaben) nur verwenden, wenn man genau weiß, was man tut und dann unbedingt leicht sperren (bei vielen Schriften ist eine zusätzliche Verkleinerung sinnvoll; *aktive* Auszeichnung).

zum Vergleich:	gesperrt	TYPOGRAFIE
	nicht gesperrt	TYPOGRAFIE

KAPITÄLCHEN: auch nur verwenden, wenn man weiß, was man tut. Das heißt, man (er)kennt den Unterschied zwischen echten und gefälschten Kapitälchen; ebenfalls leicht sperren (*integrierte* Auszeichnung).

zum Vergleich:	gesperrt	TYPOGRAFIE
	nicht gesperrt	TYPOGRAFIE

Sperren: Vergrößerung der Abstände zwischen Buchstaben eines Wortes, wobei nach Tschichold [6] gesperrte Kleinbuchstaben »immer und überall falsch« sind; Sperren ist nur etwas für »Typografie-Meister« [8].

Schriftmischung: die Verwendung einer anderen Schrift; dabei ist darauf zu achten, dass die verwendeten Schriften zueinander passen. Ist der Text in einer Serifenschrift verfasst, kann eine Serifenlose zur Auszeichnung dienen. Doch Vorsicht: Schriftmischung ist nur etwas für Experten!

Striche, Auslassungspunkte, Anführungszeichen

Da sich manche der Zeichen in der folgenden Tabelle in einer serifenlosen Schrift nicht so gut unterscheiden lassen, sind diese Zeichen im Zweifelsfall darunter noch in einer Serifenschrift dargestellt.

Bezeichnung	Zeichen	Beispiel	Erläuterung
Trennstrich	-	Silben-trennung	auch <i>Divis</i> genannt; ein kurzer Strich
Bindestrich	-	Bio- und Gentechnologie Öko-Lebensmittel	das Divis verbindet auch zusammengesetzte Wörter
Gedankenstrich	–	Ich bin immer pünktlich – bis auf wenige Ausnahmen –, was Termine angeht.	länger als das Divis, steht zwischen zwei Leerzeichen, außer in Verbindung mit Satzzeichen
Streckenstrich	–	25.09.–28.12.; auf der Strecke Bremen–Hamburg; 1–2 Telefonate	Gedankenstrich ohne Leerzeichen davor und dahinter
Minuszeichen	—	6 – 2 = 4	länger als der Bindestrich, gelegentlich dünner als der Gedankenstrich

Fortsetzung ...

... Fortsetzung

Bezeichnung	Zeichen	Beispiel	Erläuterung
Auslassungsstrich	– —	Der PC kostet 300,—€. 518,89€ 00 300,—€ —	der Gedankenstrich dient im Text auch als Auslassungszeichen; in Tabellen sollte dafür ein Geviertstrich verwendet werden, der die Breite von zwei Nullen hat
Auslassungspunkte ...		Der Bus fuhr einfach vorbei ... So ein M... aber auch.	Auslassungspunkte sollten etwas gesperrt werden; ¹ stehen sie am Satzende wird der Punkt weggelassen (Ausrufe- und Fragezeichen dürfen nicht wegfallen)
Anführungszeichen (1)	„ “ „ “	„Es wird noch regnen!“, sagte er.	nicht zu verwechseln mit den englischen Anführungszeichen (“ ” bzw. “ ”)!
Einfache Anführungszeichen (1)	, ‘ , ‘	„Dann sagte er ‚Es wird noch regnen!‘“, sagte sie.	Kennzeichnung einer Anführung innerhalb einer Anführung
Anführungszeichen (2)	» «	»Es wird noch regnen!«, sagte er.	ebenfalls in Deutschland zulässig; nicht zu verwechseln mit den französischen Anführungszeichen: « »
Einfache Anführungszeichen (2)	› ‹	›Dann sagte er ›Es wird noch regnen!‹«, sagte sie.	nicht durch die mathematischen Zeichen »kleiner als« und »größer als« ersetzen
Apostroph	, ‘ , ‘	Was für ’ne Aufregung!	nicht mit dem deutschen schließenden Anführungszeichen oder dem Fuß- bzw. Minutenzeichen verwechseln
Fuß- bzw. Minutenzeichen	, ‘ , ‘	Höhe: 5’	

Fortsetzung ...

¹Dieser Meinung sind zumindest Willberg und Forssman [8], es ist so auch im Chicago Manual of Style [3] nachzulesen. Tschichold [7] und Bringhurst [1] hingegen lehnen das Sperren von Auslassungspunkten ab.

... Fortsetzung

Bezeichnung	Zeichen	Beispiel	Erläuterung
Schrägstrich	/	WS 2000/2001; und/ oder	im Text kein Leerzeichen davor und danach

Eine Regel zu den richtigen deutschen Anführungszeichen lässt sich besser aus einer Serifenschrift ablesen: „ “. Die unteren erinnern in ihrer Form an die Zahl 99 und die oberen an die Zahl 66 („ 99 “ 66). Im Englischen ist es umgekehrt und es gibt keine unteren Anführungszeichen: “ ”.

An dieser Stelle sei auch noch etwas zu Leerzeichen vor anderen Satzzeichen gesagt. Vor folgenden Zeichen steht nie ein Leerzeichen:

. , ; : ! ? -) “ ‘

Richtig	Falsch
Dieser Satz, der zur Demonstration ein paar Satzzeichen ... enthält, hat keinen weiteren Sinn – ehrlich! (Dieser Satz auch nicht.)	Dieser Satz , der zur Demonstration ein paar Satzzeichen... enthält , hat keinen weiteren Sinn–ehrlich ! (Dieser Satz auch nicht.)

Auslassungszeichen wird dann ein Leerzeichen vorangestellt, wenn sie das Auslassen von Satzteilen markieren; sie werden direkt ohne Leerzeichen an ein Wort angehängt, wenn Teile des Wortes oder daran anschließende Satzzeichen, die ohne Leerzeichen gesetzt werden, ausgelassen werden.

Original	mit Auslassungszeichen
»Wir übergehen die Absicht, eine solche Zeile durch passendes Sperren so lang wie eine bestimmte andere zu machen. Das ist stümperhaft und soll darum hier nicht behandelt werden.« [6]	»[...] eine [...] Zeile durch passendes Sperren so lang wie eine bestimmte andere zu machen[...] ist stümperhaft [...]« [6] ²

²Hier sei jedoch angemerkt, dass zu viele Auslassungszeichen in einem Zitat an der Kontexttreue zweifeln lassen.

Abkürzungen

Wenn man Abkürzungen verwenden will, sollte man Folgendes beachten: Sparsam sein und nur übliche Abkürzungen verwenden, keine Abkürzungen am Satzanfang, keine Trennungen am Zeilenende und innerhalb von Abkürzungen Verwendung fester Abstände [2], die kleiner sind als Wortzwischenräume und »auf keinen Fall ganz wegfallen dürfen« [8]. Ein solcher Abstand wird als *Spatium* bezeichnet. Beispiele finden Sie in der Tabelle. Manche Textverarbeitungen aus Office-Paketen kennen leider kein *Spatium*. In diesem Fall sind wenigstens *feste* Abstände zu verwenden, die man mit einer Trennungssperre verbinden sollte. Dann ist auch sichergestellt, dass innerhalb von Abkürzungen nicht getrennt wird.

Richtig	Falsch	Richtig	Falsch
z. B.	z.B., z. B.	Am Zeilenende dürfen Abkürzungen wie z. B. nicht getrennt werden.	Am Zeilenende dürfen Abkürzungen wie z. B. nicht getrennt werden.

Absatzformatierung

Absätze kann man auf zwei Arten voneinander trennen: *Einzug* oder *Abstand*. Das heißt, entweder ist die erste Zeile eines neuen Absatzes bloß eingezogen, oder aber der Abstand zwischen zwei Absätzen ist erkennbar größer als der Zeilenabstand in den Absätzen. Beide Arten zu kombinieren – Absatzabstand und Zeileneinzug – ist überflüssig. Willberg und Forssman [8] schreiben dazu in Bezug auf wissenschaftliche Dokumente:

Absätze werden durch einen Einzug von ca. 4 mm gekennzeichnet. Abschnitte werden durch einen Abstand von einer Leerzeile gekennzeichnet und zum Unterschied von Absätzen ohne Einzug gesetzt. Das ist die bewährte Regel[...].

In diesem Artikel wird zur Kennzeichnung von Absätzen Abstand verwendet.

Eine Zeile sollte nicht mehr als 60 bis 70 Zeichen (einschließlich Satz- und Leerzeichen) enthalten, um die Lesbarkeit nicht unnötig zu erschweren. Will man den Platz auf einem DIN-A4-Blatt so weit wie möglich nutzen, bietet es sich an, zweispaltig zu setzen.

Die Zeilenlänge wiederum beeinflusst zusammen mit Schriftfamilie, Schriftschnitt und Schriftgröße den *Zeilenabstand*. Leider gibt es aufgrund der vielen involvierten Parameter keine Faustregel, bei welchen Einstellungen welcher Zeilenabstand geeignet ist; das kann nur das geübte Auge entscheiden. Wenn es eine Faustregel gibt, dann diese: Ein 1,5-facher Zeilenabstand (orientiert an der Bezeichnung in bekannten Textverarbeitungen) ist in normalen Texten immer zu viel!

Flattersatz

Ein gleichmäßiger sogenannter *Grauwert* ist von großer Bedeutung für die Lesbarkeit eines Textes. Zu große und ungleichmäßige Abstände zwischen Wörtern in einer Zeile oder gar zwischen Buchstaben eines Wortes (Sperrung) erschweren die Lesbarkeit eines Textes.

Flattersatz hat den großen Vorteil, dass die Wortabstände immer gleich groß sind, was positiv für einen gleichmäßigen Grauwert bzw. die Lesbarkeit ist. Andererseits wirkt Flattersatz eher unruhig, vor allem bei schlechtem Zeilenumbruch, und der (meist rechte) Flatterrand hat einen ungleichmäßigen Grauwert.

In Spalten mit weniger als 40 Zeichen kann man kaum gut lesbaren Blocksatz erzeugen, weswegen dort die Verwendung von Flattersatz sinnvoll ist. Allerdings ist guter Flattersatz nicht einfach zu erreichen. Wie beim Blocksatz sind Zeilenlänge, Sprache, in der der Text verfasst wird, der Mechanismus der Silbentrennung und des Umbruchalgorithmus⁷ der verwendeten Software von entscheidender Bedeutung.

In Abbildung 1 ist jeweils ein Beispiel für guten und weniger guten Flattersatz dargestellt.

Blocksatz

Ob man sich für oder gegen Blocksatz entscheidet, ist abhängig von der Zeilenlänge, der Sprache in der der Text verfasst wird, dem Mechanismus der Silbentrennung und dem Umbruchalgorithmus der verwendeten Software. Will man für längere Zeilen Blocksatz verwenden, muss man sicherstellen, dass die verwendete Software gleichmäßige und enge Wortabstände erzeugt; innerhalb einer Zeile sollten die Wortabstände für das Auge gleich groß sein und sich

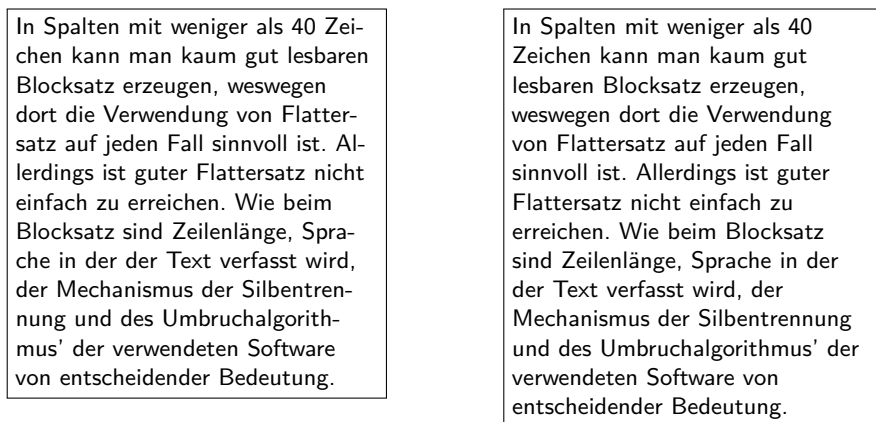


Abbildung 1: Beispiel für guten und weniger guten Flattersatz

von den Abständen der vorangehenden und nachfolgenden Zeile nicht deutlich unterscheiden. Denn ein gleichmäßiger Grauwert an den Rändern bringt nichts, wenn ansonsten innerhalb der Zeile zu große Wortabstände auftreten.

»[...] eine [...] Zeile durch passendes Sperren so lang wie eine bestimmte andere zu machen[...] ist stümperhaft [...].« [6].³

»Durch das Aufteilen des Restraums einer Zeile auf die Blocksatz-Zeilenbreite wird die Zeile gesperrt und damit zufällig, ohne jeden inhaltlichen Grund, ausgezeichnet.« [8]

Zur Verdeutlichung ist in Abbildung 2 ein Absatz mit gleichmäßigem und ein Absatz mit ungleichmäßigem Grauwert dargestellt.

Ziffern und Zahlen

Zahlen können sowohl in Ziffern als auch ausgeschrieben werden. Kleine Zahlen sollten immer ausgeschrieben werden (*zehn Personen*).⁴ Dies gilt nicht, wenn sie den Charakter einer Nummer haben (*auf Seite 8*), mit anderen in

³Zur Erinnerung: Auf Seite 24 steht das komplette Zitat.

⁴Leider gehen die Ansichten darüber, wann eine Zahl klein im hier verwendeten Sinn ist, weit auseinander. Häufig werden die Zahlen bis zwölf genannt, in Romanen findet man aber auch dreistellige und sogar Jahreszahlen ausgeschrieben.

Ob man sich für oder gegen Blocksatz entscheidet, ist abhängig von der Zeilenlänge, dem Mechanismus der Silbentrennung und dem Umbruchalgorithmus der verwendeten Software. Will man für längere Zeilen Blocksatz verwenden, muss man sicherstellen, dass die verwendete Software gleichmäßige Wortabstände erzeugt; innerhalb einer Zeile sollten die Wortabstände für das Auge gleich groß sein.

Ob man sich für oder gegen Blocksatz entscheidet, ist abhängig von der Zeilenlänge, dem Mechanismus der Silbentrennung und dem Umbruchalgorithmus der verwendeten Software. Will man für längere Zeilen Blocksatz verwenden, muss man sicherstellen, dass die verwendete Software gleichmäßige Wortabstände erzeugt; innerhalb einer Zeile sollten die Wortabstände für das Auge gleich groß sein.

Abbildung 2: Beispiel für unterschiedliche Grauwerte

Ziffern geschriebenen Zahlen verglichen werden (*von 1,5 auf 2 gestiegen*) oder mit einer abgekürzten Einheit verbunden sind (*5 min*, aber: *fünf Minuten*). Auch große Zahlen sollten ausgeschrieben werden, wenn sie nicht exakte Werte angeben (*rund tausend Zuschauer*).

Ausgeschriebene Zahlen unter einer Million werden klein und zusammen geschrieben [2]. Folgende Regeln zur Formatierung von Ziffern in Zahlen erhöhen die Leserlichkeit und erleichtern das Erfassen der Zahlen (für die Abstände sind Spatien Wortzwischenräumen vorzuziehen):

Bis zu vier Ziffern solche Zahlen werden ohne Abstände gesetzt⁵

Beispiel: 90, 900, 9000

Ab fünf Ziffern solche Zahlen »dürfen durch Abstände strukturiert werden, nicht aber durch Punkte, Apostrophe o. ä.« [8]

Beispiel: 10 000, 100 000, 1 000 000

Telefonnummern von hinten her Gruppierung in Zweiergruppen mit Abständen, wobei Vorwahl und eigentliche Rufnummer getrennt gezählt werden;

⁵Laut Duden [2] können auch vierstellige Zahlen bereits durch einen Festabstand strukturiert werden (9 000).

Durchwahlnummern werden nicht nochmal unterteilt; es kann auch mit Klammern gegliedert werden⁶

Beispiel: 02 28 / 5 64 70 08 (02 28) 5 64 70 08
02 28 / 7 01-231 (02 28) 7 01-231

Kontonummern von hinten her Gruppierung in Dreiergruppen mit Abständen

Beispiel: Konto 04 130 736

Bankleitzahlen von vorne Gruppierung in Dreiergruppen mit Abständen

Beispiel: BLZ 380 707 24

Postleitzahlen werden ohne Abstände gesetzt

Beispiel: 53121

Das scharfe S

Das scharfe S existiert nur als Kleinbuchstabe! Bei den Versalien oder Kapitälchen gibt es kein scharfes S. »GROß« ist genauso falsch wie »KLEIn«. Im Versal- oder Kapitälchensatz muss das scharfe S durch »SS« ersetzt werden [2]: GROSS bzw. GROSS. Trägt jemand ein scharfes S in seinem Namen und will dies wegen der Verwechslungsmöglichkeit auf keinen Fall durch »SS« ersetzen, kann er auch auf »SZ« ausweichen. Allerdings wird der Leser, wenn er auf »SZ« stößt, stocken, weil es ungewohnt ist.

Tabellen

Tabellen unterliegen nicht den gleichen typografischen Regeln wie Fließtext. Breite und Höhe werden von der inhaltlichen Struktur bestimmt. Bei Silbentrennungen muss auf die korrekte und sinn gerechte Trennung geachtet werden.⁷ Der wichtigste Hinweis neben dem auf korrekte Silbentrennung scheint mir jedoch der auf den Verzicht von vertikalen Linien zu sein. Diese Linien erhöhen keineswegs die Übersichtlichkeit, sondern behindern das Erfassen der

⁶Folgende Schreibweise ist ebenfalls zulässig, meines Erachtens aber weniger gut leserlich vor allem bei längeren Nummern und/oder mit internationaler Vorwahl: +49 6821 5647008

⁷Darauf explizit hinweisen zu müssen, ist dem Tabellensatz bekannter Office-Anwendungen zuzuschreiben.

Zeilen. Auf die horizontalen Linien kann man häufig auch verzichten, wenn man stattdessen einen geringen Abstand einfügt.

Schriften

Es ist sinnvoll, für längere Texte mit breiten Zeilen eine Serifenschrift zu verwenden. Die Serifen (Unterstriche) unterstützen das Auge bei der Zeilenführung und beim Zeilenrücksprung. Serifenlose Schriften – auch Groteskschriften genannt – kann man hingegen in schmalen Spalten einsetzen, in denen das Auge die Unterstützung der Serifen nicht unbedingt benötigt. Jedoch muss die Serifenlose sorgfältig ausgewählt werden, da ihre Zeichen nicht so eindeutig und unverwechselbar sind wie bei Serifenschriften. Serifenlose eignen sich auch für Beamer-Präsentationen, weil dort aufgrund der geringen Auflösung die Wirkung der Serifen verloren geht.

Anpassungen für die Schweiz

Anführungszeichen werden in der Schweiz ähnlich den französischen gesetzt, ohne Einschluss von Leerzeichen: «schweizer» Anführungszeichen («französische» Anführungszeichen). *Telefonnummern* werden von vorne beginnend in eine Dreiergruppe und anschließend in Zweiergruppen gegliedert; die dreistellige Vorwahl wird nicht weiter unterteilt: 076 / 437 28 66 oder (076) 437 28 66. *Das scharfe S* gibt es in der Schweiz nicht. An seiner Stelle wird immer »SS« bzw. »ss« gesetzt.

Zusammenfassung

Man sieht, es handelt sich – zumindest für T_EX-Anwender – um leicht umzusetzende Regeln, mit denen man insbesondere geübten Lesern das Lesen deutlich erleichtert, aber ungeübten Lesern auch nicht erschwert. Umgekehrt erschwert man geübten Lesern allerdings das Lesen sehr, wenn man diese Regeln ignoriert.

Mit den Artikeln von Marion Neubauer [4, 5] stehen L^AT_EX-Anwendern auch bereits Hilfen zur Umsetzung von Mikrotypografie-Regeln zur Verfügung. Dennoch ist ein Artikel in Arbeit, der speziell die Umsetzung der Regeln von **typokurz** (orientiert an dessen Struktur) für L^AT_EX beschreibt. Mittlerweile hat sich auch jemand gefunden, der die Umsetzung der Regeln für

OpenOffice.org beschreiben will. Angesichts der anhaltend hohen monatlichen Downloadzahlen von typokurz können wir nur alle hoffen, zukünftig vielleicht von Freunden, Bekannten, Kollegen und Studierenden auch mal schöne und gut lesbare Dokumente zu erhalten. Möglicherweise kann der eine oder andere Leser auch noch dazu beitragen, die To-Do-Liste abzuarbeiten.

Literatur

- [1] Robert Bringhurst: *The Elements of Typographic Style*; Hartley & Marks; Vancouver; 1997.
- [2] Dudenredaktion (Hg.): *Duden – Die deutsche Rechtschreibung*; Bd. 1; Dudenverlag; Mannheim · Leipzig · Wien · Zürich; 23. Aufl.; 2004.
- [3] John Grossman (Hg.): *The Chicago Manual of Style*; University of Chicago Press; Chicago; 14. Aufl.; 1998.
- [4] Marion Neubauer: *Feinheiten bei wissenschaftlichen Publikationen – Mikrotypographie-Regeln, Teil I; Die T_EXnische Komödie*; 96(4), S. 23–40; 1997; http://www.dante.de/dante/DTK//dtk96_4/Text/dtk96_4_neubauer_feinheiten.pdf.
- [5] Marion Neubauer: *Feinheiten bei wissenschaftlichen Publikationen – Mikrotypographie-Regeln, Teil II; Die T_EXnische Komödie*; 97(1), S. 25–44; 1997; http://www.dante.de/dante/DTK//dtk97_1/Text/dtk97_1_neubauer_feinheiten.pdf.
- [6] Jan Tschichold: *Erfreuliche Drucksachen durch gute Typographie*; Maro Verlag; Augsburg; 1960.
- [7] Jan Tschichold: *The Form of the Book: Essays on the Morality of Good Design*; Hartley & Marks; Vancouver; 1991.
- [8] Hans Peter Willberg und Friedrich Forssman: *Erste Hilfe in Typografie*; Schmidt; Mainz; 2001.

DIN-Brief – eine grafische Bedienoberfläche für das $\text{\LaTeX}$ -Paket *dinbrief*

Wolf-Dieter Busch

Dass $\text{\TeX}$ außerhalb des akademischen Umfelds ein Schattendasein führt, ist der Tatsache geschuldet, dass es formal eine Programmiersprache ist. Schade eigentlich.

Einleitung

Die Universität Karlsruhe hat ein $\text{\LaTeX}$ -Paket namens **dinbrief** entwickelt, dessen Makros nicht nur für ausgewogene Typografie, sondern auch für Maßhaltigkeit und vorschriftsmäßige Formfelder sorgen. Das Paket **dinbrief** sollte in keinem Haushalt fehlen. Tut es aber doch. Das sollten wir ändern.

DIN-Brief

Das Tcl-Script *dinbrief* ist eine Mini-Anwendung für das $\text{\LaTeX}$ -Paket *dinbrief*. Es ist ein einfaches Formularfenster – ähnlich einem HTML-Formular – in dem für Absender, Empfänger, Betreff, Haupttext ein schlichtes Textfeld angelegt ist.

Das Tool ist auf intuitive Benutzung optimiert. Wenn der Haupttext strukturiert wird, werden (im richtigen Leben) diese Stilmittel verwendet:

- Aufzählungen, um die Strukturen zu verdeutlichen;
- dito Nummerierungen;
- Tabellen zur besseren Übersicht;
- Unterbetriffs, um lange Brieftexte in Sinneinheiten zu untergliedern.

Was muss der Anwender von Din-Brief dazu lernen? Nahezu nichts:

Wenn mehrere Absätze mit Nummern beginnen, erkennt *dinbrief* dies automatisch und formatiert sie entsprechend.

Wenn der Anwender die Tab-Taste drückt, produziert dinbrief vollautomatisch eine Tabellen-Umgebung, selbstverständlich mit automatischer Nachführung der Spaltenbreite und Kommaabündigkeit bei reinen Zahlenspalten.

Für Unterbetriffs wird der Anfang des Absatzes mit Sternchen oder Schräg- oder Unterstrichen umgeben. Dinbrief macht daraus eine Hervorhebung.

Dass einfache Zeilenwechsel zum Makro `\` und der mehrfache Zeilenwechsel zum Makro `\par` führen, versteht sich von selbst.

Selbst in diesem begrenzten Nutzbereich kann es vorkommen, dass eine Textstelle auf elektrischem Wege gesucht werden muss. Dafür gibt es eine inkrementelle Suche ohne Schnickschnack.

Für Feinjustierung sowie Export nach L^AT_EX und PDF gibts ein Menü. Richtige kleine Sekretärin, das Teil, gell?

Technisches

Die Datei **dinbrief.zip** enthält das Tool in drei Ausführungen:

- als ausführbares Windows-Programm (**dinbrief.exe**);
- als Sammlung von Tcl-Scripten, aufzurufen mit **dinbrief.sh** bzw. **dinbrief.bat**, verlangt natürlich nach einer Tcl/Tk-Installation;
- als Tcl-Kit.

Installation: Gewünschte Datei in irgendein Verzeichnis stellen. Sorge tragen, dass T_EX und ein PDF-Betrachter verfügbar sind. Bei Erstaufruf mit Menü einstellen, wo pdfL^AT_EX und der PDF-Betrachter zu finden sind.

Deinstallation bei Nichtgefallen: Verzeichnis löschen reicht.

Eine Systembeschreibung ist unter <http://tinyurl.com/afukm> verfügbar.

Zusammenfassung

Es gibt viele Probleme, für die sich T_EX anbieten würde. Eine freundliche grafische Bedienoberfläche erlaubt es dem Benutzer, sich voll auf seine Aufgabe zu konzentrieren. Das Tool dinbrief ist ein Beispiel von vielen (noch ungeschriebenen, aber hoffentlich kommenden) GUIs für L^AT_EX.

Minimal? – Vom Testen eines Textsatzsystems

Gerd Neugebauer

Ein System wie $\epsilon\chi\text{T}_{\text{E}}\text{X}$ muss getestet werden. Dabei kommt die Kunst der Minimalbeispiele zum Einsatz. Diese treten hier in der Form von Testfällen auf. $\epsilon\chi\text{T}_{\text{E}}\text{X}$ soll durch Testfälle so abgesichert werden, dass während der Weiterentwicklung keine ungewollten Veränderungen an den implementierten Funktionalitäten auftreten.

Einleitung

In Mailing-Listen wie TEX-D-L und News-Gruppen wie de.comp.text.tex wird auf Fragen immer wieder mit einem Hinweis auf ein Minimalbeispiel geantwortet. Ein Minimalbeispiel dient dabei dazu, die Essenz des Effekts, den man zeigen will, herauszuarbeiten. Das hilft auf der einen Seite zu verstehen, welche Elemente daran beteiligt sind – wenn man noch etwas weglassen kann, ist es noch nicht minimal. Auf der anderen Seite erlaubt es anderen Anwendern, den Effekt zu reproduzieren. Damit kann nachgewiesen werden, dass der Effekt immer noch auftritt, auch wenn ein anderes $\text{T}_{\text{E}}\text{X}$ -System benutzt wird.

Das Konzept des Minimalbeispiels ist auch noch an anderer Stelle relevant, nämlich in der Software-Entwicklung. Hier kommt es in der Form von Testfällen vor. Eine Software sollte normalerweise eine Reihe von Anforderungen erfüllen. Um dies zu testen, werden die Anforderungen in Testfälle gegossen. Damit hat man eine – im besten Fall – große Liste von Testfällen, die es gestattet detailliert abzuschätzen, in welchem Maß die Anforderungen erfüllt sind.

Im Gegensatz dazu steht ein einzelner großer Testfall. Entweder er geht durch, oder er geht eben nicht durch. Hier ist eine detaillierte Bewertung schwieriger. Auch ist es schwieriger, einen Fehler zu lokalisieren und gegebenenfalls zu beheben.

Während der Entwicklung können Testfälle dabei helfen, das Funktionieren von Code zu beurteilen. Oftmals wird Code geändert – sei es, dass er um neue Funktionalität erweitert wird, oder sei es, dass er verbessert wird. Das wird heute oft als Refactoring bezeichnet [3]. Zum Extrem wird dies beim

Test-Driven Development getrieben [1, 6]. Hier werden als erstes die Testfälle erstellt und danach die Software implementiert.

In $\mathcal{NTS}$ wurde im Gegensatz zu dem bisher Gesagten im Wesentlichen ein großer Testfall benutzt – das $\text{\TeX}$ Book [5]. Das zeigt zwar viele Eigenschaften von $\text{\TeX}$, aber eben nicht alle. Und wenn dann ein Fehler auftritt, kann man nur schwer beurteilen, wie schwerwiegend er ist.

Auch der Trip-Test von $\text{\TeX}$ testet einige Eigenschaften ab. Dies betrifft vor allem das Verhalten von $\text{\TeX}$ in Randbereichen des Einsatzes. Es ist denkbar, ein relativ einfaches Programm zu schreiben, das den Trip-Test bestehen würde, aber damit noch bei weitem nicht geeignet, damit Texte zu setzen. Auf der anderen Seite werden im Trip-Test auch Eigenschaften abgetestet, die für ein $\text{\TeX}$ -kompatibles System nicht relevant sind.

$\epsilon\lambda\text{\TeX}$ (<http://www.extex.org>) hat als ein Ziel, in einem Kompatibilitätsmodus die gleiche Funktionalität wie $\text{\TeX}$ anzubieten. Deshalb ist es sinnvoll, hier eine Test-Suite aufzubauen, die sowohl die Kompatibilität sicherstellt, wie auch die Stabilität des bereits erstellten Codes überprüft.

Hyphenation – ein Beispiel

Um das Testen etwas anschaulicher zu gestalten, betrachten wir ein Beispiel aus dem Bereich der Silbentrennung.

Testfälle entstehen aus Anforderungen. Anforderungen an $\text{\TeX}$ finden sich in [4, 5]. Also liest man, was dort steht und versucht, sich etwas auszudenken, wie es überprüft werden kann. Selbst ohne allzu große Kenntnisse in $\text{\TeX}$ kann man auf diese Weise Schritt für Schritt die Anforderungen erfassen und in Testfälle übersetzen.

Beipielsweise findet sich in [4] die folgende Aussage:¹

902. $\text{\TeX}$ will never insert a hyphen that has fewer than `\lefthyphenmin` letters before it or fewer than `\righthyphenmin` after it; hence, a short word has comparatively little chance of being hyphenated.

¹Übersetzung: $\text{\TeX}$ wird keine Trennstelle einfügen, die weniger als `\lefthyphenmin` Buchstaben davor oder weniger als `\righthyphenmin` danach hat; damit hat ein kurzes Wort eine relativ geringe Chance, getrennt zu werden.

Die Silbentrennung in $\text{\TeX}$ wird dann benutzt, wenn der aktuelle Absatz nicht ohne Trennstellen »gut« umbrochen werden kann. In diesem Fall werden zusätzliche, optionale Trennstellen eingefügt. Dazu gibt es einen Algorithmus, der aus `\pattern{}` gespeist wird. Zusätzlich gibt es noch die Ausnahmen, die über `\hyphenation{}` eingegeben werden.

Für den ersten Testfall wollen wir uns auf die Werte aus `\hyphenation` konzentrieren. Gilt die Aussage auch für diese oder nur für die Muster-basierten Trennstellen?

Um das zu überprüfen, brauchen wir ein Minimalbeispiel. Da wir wirklich minimalistisch arbeiten wollen, nutzen wir zuerst einmal nur `ini\TeX`. Dies ist der eigentliche Programmkern, ohne dass irgendwelche Definitionen aus Formaten gelesen werden.

Damit müssen wir uns um fast alles auch wirklich selbst kümmern. Das fängt damit an, dass die Klammern noch nicht als solche behandelt werden – die Kategorien müssen erst mit `\catcode` eingestellt werden. Auch gibt es noch keinen Font. Also laden wir `cmr10` und benutzen ihn im Folgenden. Für diesen Font stellen wir auch noch den Hyphen-Charakter ein. Der Wert ist der ASCII-Code des Zeichens, das bei der Trennung eingefügt werden soll. Der Wert 45 ist dabei der Code für das Zeichen »-«. Das führt zu folgenden Zeilen Code:

```
\catcode'\{=1
\catcode'\}=2
\font\font cmr10 \font
\hyphenchar\font=45
```

Jetzt können wir mit dem eigentlichen Test beginnen. Wir wollen einem Wort mit `\hyphenation` eine Trennstelle zuweisen. Da es nicht relevant ist, welches Wort wir nehmen, haben wir uns für `abcdef` entschieden, das in der Mitte trennbar sein soll. Damit auch getrennt werden muss, stellen wir die Seitengröße auf eine Breite von 25 Punkten und eine Höhe von 100 Punkten. Schließlich setzen wir noch `\lefthyphenmin` und `\righthyphenmin` jeweils auf 0, damit auch wirklich getrennt werden kann.

```
\hyphenation{abc-def}
\hsize=25pt
\vsize=100pt
\lefthyphenmin=0
\righthyphenmin=0
```

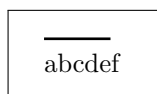
Um die Breite zu sehen, zeichnen wir mit `\hrule` eine Linie der Textbreite und können nun unser Wort »abcdef« ausgeben. Mit `\end` schließen wir die Verarbeitung ab.

```
\hrule width \hsize height 1pt  
  
abcdef  
  
\end
```

Wenn wir die obigen Zeilen in einer Datei `htest.tex` haben, können wir auch schon ansehen, was $\text{\TeX}$ daraus macht:

```
initex htest.tex
```

Im Preview sehen wir dann in etwa das Folgende – allerdings ohne den Rahmen:

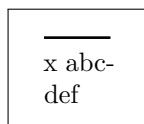


Das Wort wurde nicht getrennt, obwohl wir das erwartet hatten. Irgend etwas stimmt da nicht. Vielleicht wird die Regel doch nicht auf Worte aus `\hyphenation` angewendet?

Doch halt: Da war doch noch etwas. $\text{\TeX}$ trennt niemals das erste Wort eines Absatzes. Also müssen wir dafür sorgen, dass unser Wort nicht das erste ist. Dazu schreiben wir einfach ein x davor:

```
x abcdef
```

Damit erhalten wir:

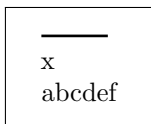


Genau das ist der erwartete Effekt. Also haben wir unser Minimalbeispiel erreicht. Daraus können wir noch einige weitere Testfälle ableiten, indem wir `\lefthyphenmin` und `\righthyphenmin` variieren oder den Bindestrich nach links oder rechts verschieben.

Dann ergibt beispielsweise der folgende Code das Ergebnis, dass keine Trennung stattfindet:

```
\lefthyphenmin=4
\righthyphenmin=4
```

Das sieht dann folgendermaßen aus:



Die Test-Umgebung

Die Test-Suite von $\epsilon\chi\text{TeX}$ hat das Ziel, die Testfälle automatisiert ablaufen lassen zu können. Damit wird es möglich, ohne menschlichen Eingriff feststellen zu können, in welchem Maß die Anforderungen schon erfüllt sind. Und schließlich ist es bei einer größeren Anzahl von Testfällen aufwändig, jedes Mal im Preview ein Ergebnis zu überprüfen.

$\epsilon\chi\text{TeX}$ wird in Java entwickelt. Damit liegt es nahe, die Werkzeuge aus der Java-Welt einzusetzen. Als ein Standard-Werkzeug für Unit-Tests hat sich hier JUnit [2] durchgesetzt. JUnit ist im Kern recht einfach – sowohl was die dahinter liegenden Konzepte als auch, was die Anwendung betrifft. Jeder Testfall ist eine Java-Methode; die umschließende Java-Klasse stellt eine Test-Suite dar.

Damit das funktioniert, muss die Klasse von einer Basisklasse abgeleitet sein und die Methoden müssen einer Namenskonvention genügen. Die Basisklasse ist `junit.framework.TestCase`, und der Klassenname muss mit »Test« enden. Die Konvention sagt, dass die Methoden `public void` sein müssen und mit der Zeichenkette »test« beginnen.

In den Testfall-Methoden sollte dann auf einige der vielen »assert«-Methoden der Basis-Klasse zurückgegriffen werden. Diese dienen sowohl dazu, Nachbedingungen zu überprüfen, als auch, dem Test-Framework Erfolg oder Misserfolg zu signalisieren. Das sieht beispielsweise folgendermaßen aus:

```
public class MyTest extends TestCase {

    public MyTest(final String name) {
```

```

    super(name);
}
public void test1() throws Exception {
    // Perform tests here
    assertEquals(2, 1 + 1);
}
}

```

So viel als kurzer Ausflug in die »Tiefen« von Java. Testfälle kann man aber auch erstellen und laufen lassen, wenn man kein Java kennt und nur die passenden Werte an die richtigen Stellen schreibt.

Die Aufgabe für $\epsilon\text{\textit{x}}\text{\textit{T}}\text{\textit{E}}\text{\textit{X}}$ ist vielfach die gleiche: Der Interpreter wird mit einer Eingabe gefüttert. Die Ausgabe ist einerseits der Strom der »Nodes«, der zur Ausgabe geht und der Fehler-Kanal, der am Ende in der Log-Datei landet. Diese beiden Ströme können abgefangen und mit vorgegebenen Werten verglichen werden. Dafür gibt es die Klasse `ExTeXLauncher` und die Methode `assertOutput`. Dieser Methode gibt man drei Zeichenketten mit:

- den Code, der ausgeführt werden soll,
- das Ergebnis im Ausgabekanal und
- das Ergebnis in der Log-Datei.

Das sieht dann beispielsweise folgendermaßen aus:

```

public class RelaxTest extends ExTeXLauncher {

    public RelaxTest(final String name) {
        super(name);
    }
    public void test1() throws Exception {
        assertOutput(
            // --- Input code
            "\\relax\\end",
            // --- Log channel
            "",
            // --- Output channel
            ""
        );
    }
}

```

Man muss nur wissen, dass man den Backslash in der Zeichenkette verdoppeln muss und andere Sonderzeichen durch Voranstellen eines Backslash kenntlich gemacht werden müssen. Insbesondere der String-Begrenzer (\\") und das Newline-Zeichen (\\n) sind zu nennen.

Das war alles. Schon kann man die Tests laufen lassen. Damit alles noch mit einer netten grafischen Oberfläche versehen wird, genügen die folgenden drei Zeilen:

```
public static void main(final String[] args) {
    junit.swingui.TestRunner.run(RelaxTest.class);
}
```

Das Ergebnis sieht dann wie das Beispiel in Abbildung 1 aus. Hier wird angezeigt, wieviele Testfälle enthalten sind und wieviele erfolgreich oder mit einem Fehler beendet wurden. Der Gesamterfolg wird durch den grünen Balken signalisiert; anderenfalls wäre er rot. In einem solchen Fall gilt es dann herauszufinden, wo die Ursache für den Fehlschlag zu suchen sein könnte.

Verschwiegen haben wir bisher, wie man die Tests laufen lassen kann. Und auch das geht recht einfach. Hierfür ist in $\epsilon\chi\text{TeX}$ das Skript `develop/junit` vorgesehen. Hier werden alle Einstellungen passend vorgenommen und der Test wird gestartet:

```
develop/junit de.dante.extex.interpreter.primitives.RelaxTest
```

Das funktioniert auf die gleiche Weise sowohl unter Unix wie auch unter Windows. Man muss also kein Unix-affiner Hacker sein, um Testfälle für $\epsilon\chi\text{TeX}$ erstellen und ausführen zu können.

Genauso gut lassen sich die Testfälle aber auch – ohne diese Zusatzzeilen – aus einer IDE wie Eclipse (<http://www.eclipse.org>) heraus abarbeiten. Das ist an dem gleichen Beispiel in Abbildung 2 zu sehen.

Schließlich gibt es auch noch die textbasierte Variante. Eine Ausgabe kann dann wie in Abbildung 3 aussehen. Diese wird auch benutzt, um alle Testfälle en bloc abarbeiten zu lassen.

Zusammenfassung und Ausblick

Im Rahmen von $\epsilon\chi\text{TeX}$ wurde bisher eine Test-Suite in einer Basis-Version erstellt. Diese enthält bereits über 7000 Testfälle. Da die Testfälle immer nur

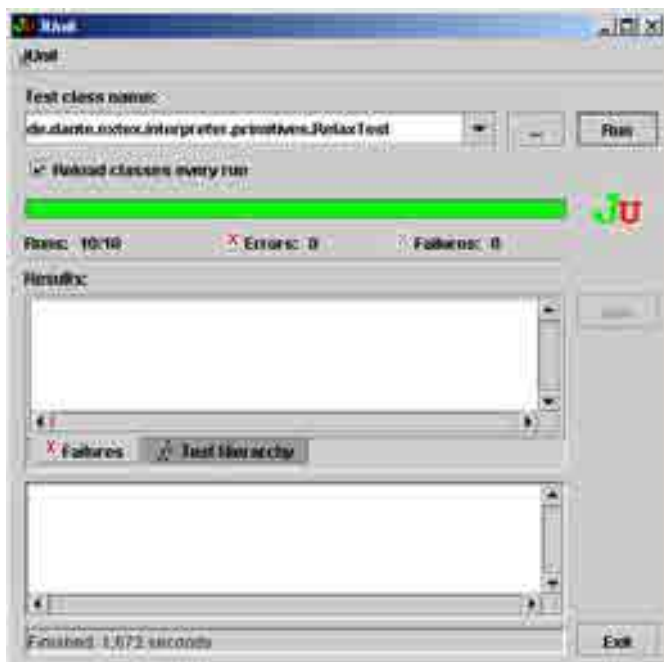


Abbildung 1: Ein Test-Bericht von JUnit

einen kleinen Aspekt überprüfen, ist zu erwarten, dass es sehr viele davon geben muss. Eine einigermaßen vollständige Test-Suite wird sicher auf mehr als die doppelte Größe des bisher Erreichten anwachsen müssen.

Wer sich bisher in der Kunst des Minimalbeispiels auf der Ebene von $\text{\LaTeX}$ versucht hat, kann sich hier in einem neuen Minimalismus üben. Wegen des tief gewählten Startpunkts muss man auch nicht viele Pakete kennen. Dafür muss man sich um einige Dinge kümmern, die sonst schon geregelt sind. Auf diese Weise kann man sehr einfach tiefe Einblicke in die Arbeitsweise von $\text{\TeX}$ – und $\epsilon\text{\TeX}$ – erhalten.

Für $\epsilon\text{\TeX}$ wäre es sehr hilfreich, wenn sich der eine oder andere finden würde, der sich in solchen Minimalbeispielen üben will und die Ergebnisse für $\epsilon\text{\TeX}$ beisteuert. Eine Mitarbeit in diesem oder einem anderen Bereich von $\epsilon\text{\TeX}$ ist jederzeit willkommen.



Abbildung 2: JUnit in Eclipse

```
.....  
Time: 10,054  
  
OK (10 tests)
```

Abbildung 3: JUnit, textbasiert

Literatur

- [1] Kent Beck: *Test-Driven Development*; Addison-Wesley; Boston, MA; 2003.
- [2] Kent Beck: *JUnit kurz & gut*; O'Reilly; Köln; 2005.
- [3] Martin Fowler: *Refactoring – Wie Sie das Design vorhandener Software verbessern*; Addison-Wesley; 2000.
- [4] Donald E. Knuth: *T_EX – the Program*; Bd. B von *Computers and Typesetting*; Addison-Wesley; Boston, MA; 1986.
- [5] Donald E. Knuth: *The T_EXbook*; Bd. A von *Computers and Typesetting*; Addison-Wesley; Boston, MA; 1996.
- [6] Johannes Link: *Unit Tests mit Java – der Test-First-Ansatz*; dpunkt.verlag; Heidelberg; 2002.

Tipps und Tricks: Animierte GIF-Grafiken mit L^AT_EX

Rolf Niepraschk

In diesem Artikel wird gezeigt, wie man mit L^AT_EX animierte GIF-Grafiken erzeugen kann. Die Einzelgrafiken, die später die Animation ergeben sollen, werden unter Verwendung des Makro-Pakets `pstricks` [4] gezeichnet.

Das folgende Beispieldokument bildet die Grundlage für die weiteren Schritte:

```
1 \documentclass[12pt]{article}      % anim-demo.tex
2 \usepackage[dvipsnames,prologue]{pst-pdf}
3 \usepackage{multido,calc}
4
5 \newcommand*\maxstep{17}
6
7 \begin{document}
8
9 \setlength\unitlength{1in/300}% 300dpi
10 \psset{unit=\unitlength}
11
12 \multido{\r=0+22.5,\i=1+1}{\maxstep}{%
```

```

13 \begin{pspicture}(-300,-300)(300,300)
14   \pswedge[fillstyle=solid,fillcolor=Apricot]{300}{0}{\r}
15   \sffamily\bfseries\large\rput(250,250){\i}
16 \end{pspicture}
17 }
18
19 \end{document}

```

Das genutzte Paket `pst-pdf` [3] lädt die Pakete `pstricks` und `preview`. Letzteres ist dafür zuständig, dass der Inhalt jeder einzelnen der 17 `pspicture`-Umgebungen formatfüllend als komplette Seite in die Ausgabedatei geschrieben wird.

```

latex anim-demo.tex
dvips -o anim-demo.ps anim-demo.dvi

```

Die aus der DVI-Datei erzeugte PostScript-Datei muss nun in einzelne Pixelgrafiken gewandelt werden, was mit Hilfe des Programms »`ghostscript`« [1] geschieht:

```

gs -q -dNOPAUSE -dBATCH -dGraphicsAlphaBits=4 -dTextAlphaBits=4 \
-sDEVICE=png256 -sOutputFile=anim-demo%03d.png anim-demo.ps

```

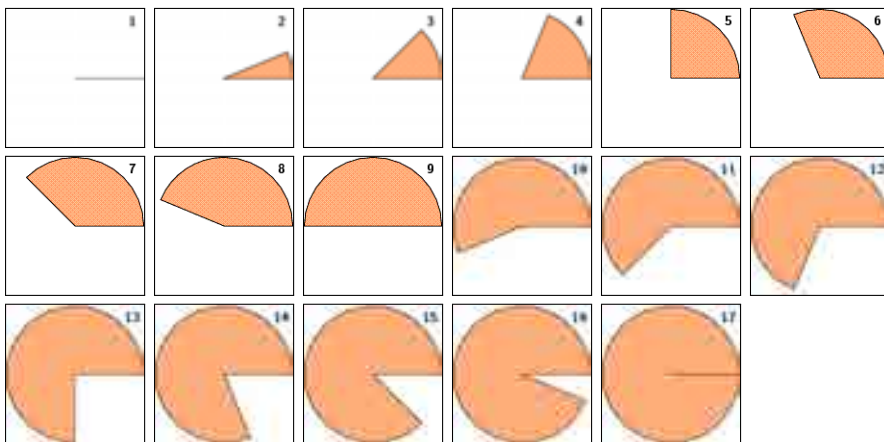


Abbildung 1: Die einzelnen Phasen der Animation (alle PNG-Dateien)

Die Angabe `%03d` sorgt dafür, dass alle resultierenden PNG-Dateien fortlaufend 3-stellig nummeriert werden. Je nach gewünschter Farbauflösung

kann als `DEVICE` auch `png16` oder `pnggray` gewählt werden. Die Optionen `GraphicsAlphaBits=4` und `TextAlphaBits=4` sorgen dafür, dass mit Hilfe des sogenannten »Antialiasing« die Darstellung von Kurvenzügen bei grafischen Bestandteilen bzw. Schriftzeichen verbessert wird. Im nächsten Schritt werden alle Einzelgrafiken kombiniert in das GIF-Format konvertiert. Dies geschieht mit dem Programm »convert« [2]:

```
convert -delay 100 anim-demo???.png anim-demo.gif
```

Die Angabe `-delay` bestimmt die Zeit, während der ein Einzelbild zu sehen ist. Sie wird als Vielfaches von 0,01 s angegeben.

Zur Wiedergabe der GIF-Animation sind beispielsweise die meisten der aktuellen Web-Browser geeignet.

Literatur und Software

- [1] *Ghostscript*; 2006; <http://www.ghostscript.com/>.
- [2] *ImageMagick*; 2006; <http://www.imagemagick.org/>.
- [3] Hubert Gäßlein, Rolf Niepraschk: *The pst-pdf package*; Juni 2005; CTAN: macros/latex/contrib/pst-pdf/pst-pdf.dtx.
- [4] *PSTricks*; 2006; <http://www.pstricks.de/>.

T_EX-Beiprogramm

Neue Pakete auf CTAN

Jürgen Fenn

Der Beitrag stellt neue Pakete auf CTAN seit Mitte Januar 2006 bis zum Redaktionsschluss vor. Die Liste folgt der umgekehrten chronologischen Reihenfolge.

Schlichte Updates werden nicht aufgeführt. Diese können auf der moderierten *tex-announce*-Mailingliste verfolgt werden, die auch unter <http://blog.gmane.org/gmane.comp.tex.ctan.announce> im WWW verfügbar ist.

digiconfigs hilft beim Auszeichnen von »Konfigurationen«, also Matrizen aus `\circ`'s und `\bullet`'s. Autor ist *Lars Madsen*.

CTAN:macros/latex/contrib/
digiconfigs

hrlatex stellt ein Standard-Setup für kroatische L^AT_EX-Anwender zur Verfügung. Autor ist *Marcel Maretic*.

CTAN:language/croatian/
hrlatex

pkfix-helper ist ein Programm von *Scott Pakin*, das dazu dient, Bitmap-Fonts, die *dvips* (ab v5.58) in PostScript-Dateien einbindet, durch Fonts zu ersetzen, deren

Darstellung von der Auflösung unabhängig sind.

CTAN:support/pkfix-helper

dvdcoll ist ein neues Paket, das beim Erstellen von Verzeichnissen für DVD-Sammlungen helfen will. Autor ist *Josef Kleber*.

CTAN:macros/latex/contrib/
dvdcoll

thailatex von *Theppitak Karoonboonyanan* dient zum Setzen von thailändischen Texten mit den L^AT_EX-Standardklassen und soll ein Teil von *babel* werden. Dazu die passenden skalierbaren Fonts.

CTAN:language/thai/thailatex
 CTAN:fonts/thai/thaifonts-
 scalable

orderer, ein Perl-Skript, ordnet die Literaturfundstellen in der Reihenfolge, in der sie in einem Text auftreten, wie von vielen Zeitschriften verlangt, ähnlich dem Paket *orderrefs*. Autor ist *M. Siddikov*.

CTAN:support/orderer

ifxetex von *Will Robertson* entspricht sozusagen dem Paket *ifpdf* für *pdf_{tex}*: Befehle werden abhängig davon ausgeführt, ob Xe_{TeX} verwendet wird oder nicht. Funktioniert mit *La_{TeX}* und Plain *TeX* mit *ε-TeX*-Erweiterungen.

CTAN:macros/xetex/latex/
 ifxetex

sanhyph sind Unicode-Trennmuster für die Sanskrit- und Pankrit-Sprachen Devanagari, Bengali, Kannada, Malayalam und Telugu von *Yves Codet*.

CTAN:macros/xetex/
 hyphenation/sanhyph

xltxtra bietet Zugaben und *Bugfixes* zur Verwendung von Xe_{TeX}, das mittlerweile auch auf Linux und Windows portiert worden ist. Autor ist *Will Robertson*.

CTAN:macros/xetex/latex/
 xltxtra

grchyp von *Yves Codet* stellt griechische Trennmuster für Xe_{TeX} bereit.

CTAN:macros/xetex/
 hyphenation/grchyp

enpassant von *Ulrike Fischer* enthält alle Dateien, die man braucht, um die Schachfonts von www.enpassant.dk mit (La)TeX zu verwenden.

CTAN:fonts/chess/enpassant

svn-multi hilft beim Einfügen von Stichworten aus dem System *subversion* zur Verwaltung verschiedener Versionen von Dateien an beliebiger Stelle in einem *La_{TeX}*-Dokument. Im Gegensatz zu dem sehr ähnlichen Paket *svn* kann man *svn-multi* auch bei Dokumenten verwenden, die auf mehrere Dateien aufgeteilt wurden. Autor ist *Martin Scharrer*.

CTAN:macros/latex/contrib/
 svn-multi

mrcheckbib ist ein Tool, das jeden Eintrag einer *bibtex*-Datei anhand der *AMS-MRef*-Datenbank überprüft. Autor ist *John Cagnol*.

CTAN:biblio/bibtex/utis/
 mrcheckbib

Free Math Font Survey von *Stephen Hartke* bietet eine Übersicht der unter *TeX* und *La_{TeX}* frei verfügbaren Schriften für den ma-

thematischen Formelsatz, einschließlich Verwendungsbeispielen, Downloadlinks und Angaben zu den L^AT_EX-Paketen, um die Fonts zu laden.

CTAN:info/Free_Math_Font_Survey

spotcolor hilft bei der Verwendung von Schmuckfarben/Sonderfarben,¹ wenn man eine PDF-Datei mit *pdflatex* erzeugt. Autor ist *Jens Elstner*.

CTAN:macros/latex/contrib/spotcolor

lewis dient zum Zeichnen von Lewis-Strukturformeln, beschränkt auf Elemente, für die die Oktettregel gilt. Autor ist *Kent Horvath*.

CTAN:macros/latex/contrib/lewis

pst-pdgr stellt ein Interface zur Verfügung, um medizinische Stammbäume mit L^AT_EX und PlainT_EX in einer *pspicture*-Umgebung zu zeichnen. Mit dem Perl-Skript *pedigree-perl* können Daten aus einer Tabellenkalkulation übernommen werden. Autor ist *Boris Veytsman*.

CTAN:graphics/pstricks/contrib/pedigree/pst-pdgr
CTAN:graphics/pstricks/contrib/pedigree/pedigree-perl

bghyphen sind neue Trennmuster für Texte in bulgarischer Sprache zur Verwendung mit *babel*. Autor ist *Georgi Boshnakov*.

CTAN:language/hyphenation

rfl von *Patrick Gundlach* will beim Installieren von Fonts helfen. Das Paket umfasst einige Programme und Bibliotheken, die in Ruby geschrieben wurden, um *tfm*-, *afm*-, *vf*- und *enc*-Dateien zu lesen und zu schreiben sowie um Font-Metriken zu erstellen. Die Dokumentation zu dem noch experimentellen Release findet man unter <https://foundry.supelec.fr/projects/rfl/>.

CTAN:support/rfil

hyperxmp dient dazu, Daten im Adobe-Format XMP (*eXtensible Metadata Platform*) in einem L^AT_EX-Dokument einzubinden, wenn man eine PDF-Datei erzeugt. Das Paket arbeitet mit dem *hyperref*-Paket zusammen. Unterstützt werden *pdflatex*, *latex+dvipdfm* und *latex+dvips+ps2pdf*. Autor ist *Scott Pakin*.

CTAN:macros/latex/contrib/hyperxmp

cms4talks von *Olle Nebendahl* ist ein Content Management System. Mit dem Java-Programm

¹Vgl. <http://de.wikipedia.org/wiki/Schmuckfarbe> .

kann man Vorträge und Präsentationen, die in $\text{\LaTeX}$ geschrieben wurden, in einer MySQL-Datenbank durchsuchen und neu anordnen.

CTAN:support/cms4talks

opteng von *Patrick Egan* beinhaltet eine Vorlage und einen `sty` zum Abfassen von Aufsätzen für die SPIE-Zeitschriften *Optical Engineering* und *Optical Engineering Letters*. Weil bei den *Letters* ein striktes Limit auf drei Seiten besteht, kann mit diesem Paket die Seitenzahl des Texts abgeschätzt werden.

CTAN:macros/latex/contrib/
opteng

tabto ist ein neues Paket von *Donald Arseneau*, mit dem man innerhalb eines Absatzes feste Tabulatoren setzen kann.

CTAN:macros/latex/contrib/
tabto

Beamer tutorial bietet eine portugiesische Einführung in das Paket *beamer*. Autor ist *Carlos Campani*.

CTAN:info/portuguese/beamer

titlepage-uni-dortmund stellt das Layout für die Titelseite einer Arbeit an der Universität Dortmund bereit. Autor ist *Gerd Sebastiani*.

CTAN:macros/latex/contrib/
titlepage-uni-dortmund

cursolatem von *Carlos Campani* ist eine kurze Präsentation zur Einführung in $\text{\LaTeX}$ in portugiesischer Sprache.

CTAN:info/portuguese/
cursolatem

xypic-tutorial ist ein Tutorium zu dem Paket *xypic* in portugiesischer Sprache. Autor ist *Carlos Campani*.

CTAN:info/xypic-tutorial

mbkangtex enthält ein Python-Skript, das einen bengalischen Text, der mit BangTeX geschrieben wurde, in einen Text mit lateinischen Buchstaben konvertiert. Autor ist *Koushik Ray*.

CTAN:language/bengali/
mbkangtex

thuthesis von *Ruini Xue* bietet eine Vorlage für Abschlußarbeiten der Universität Tsinghua.

CTAN:macros/latex/contrib/
thuthesis

chessboard ist ein neues Paket von *Ulrike Fischer*, mit dem man Spielstellungen von Schachpartien setzen kann. Auch Ausschnitte eines Schachbretts können gezeichnet werden, auch Farben werden unterstützt.

CTAN:macros/latex/contrib/
chessboard

sort-by-letters enthält eine Sammlung von Bib_TE_X-Stilen auf der Basis von *natbib*, mit denen man die Einträge einer Bibliographie alphabetisch nach dem Namen des Autors oder des Herausgebers sortiert ausgeben lassen kann. Autor ist *Thomas van Oudenhove*.

CTAN:biblio/bibtex/contrib/
sort-by-letters

boxhandler von *Steven B. Segletes* hilft dabei, die Beschriftungen von Gleitobjekten (*captions*) zu gestalten. Sie können auch ganz unterdrückt werden.

CTAN:macros/latex/contrib/
boxhandler

AddTeX2Eps bietet ein Paket zur Bearbeitung von EPS-Grafiken aus *Mathematica* mit L_AT_EX-Befehlen. Autor ist *Janko Slavic*.

CTAN:graphics/AddTeX2Eps

alphabib ist ein Bash-Script, mit dem man alphabetische Überschriften in eine bbl-Datei einfügen kann, was vor allem zur schnelleren Orientierung nach dem Anfangsbuchstaben des Eintrags in umfangreichen Bibliographien sinnvoll sein kann. Autor ist *Mael Hilléreau*.

CTAN:biblio/bibtex/utls

noitcrul von *Paul Ebermann* definiert eine Variante des `\underline` Befehls ohne *Italics*-Korrektur für den Mathematik-satz.

CTAN:macros/latex/contrib/
noitcrul

extpfeil hilft beim Zeichnen von Pfeilen und beim Definieren neuer Befehle hierzu. Autor ist *Paul Ebermann*.

CTAN:macros/latex/contrib/
extpfeil

dateiliste von *Paul Ebermann* übernimmt eine Liste mit allen L_AT_EX-Files, mit denen ein Dokument kompiliert worden ist, in Form einer Tabelle in das Dokument, einschließlich CVS-Info.

CTAN:macros/latex/contrib/
dateiliste

faktor dient zum Setzen von Quotientenstrukturen wie *A modulo B*. Autor ist *Paul Ebermann*.

CTAN:macros/latex/contrib/
faktor

randbild von *Paul Ebermann* dient dazu, kleine Plots von Kurven, die man mit *pst-plot* erzeugt, in den Seitenrand zu setzen.

CTAN:macros/latex/contrib/
randbild

robustcommand bietet eine Variante von `\DeclareRobustCommand`, die

kontrolliert, ob der Befehl schon früher definiert worden war. Autor ist *Paul Ebermann*.

CTAN:macros/latex/contrib/
robustcommand

pauldoc ist ein Paket, das man benötigt, um die Dokumentation zu den anderen Paketen zu kompilieren, die *Paul Ebermann* auf CTAN bereitgestellt hat.

CTAN:macros/latex/contrib/
pauldoc

thesis-titlepage-fhac von *Jürgen A. Lamers* ist der Stil für die Titelseite von Diplomarbeiten der FH Aachen (Standort Jülich).

CTAN:macros/latex/contrib/
thesis-titlepage-fhac

rsc.bst ist ein BibTeX-Stil für die meisten RSC-Zeitschriften. Autor ist *Joseph Wright*.

CTAN:biblio/bibtex/contrib/
misc/rsc.bst

jneurosci ist ein neuer BibTeX-Stil für das *Journal of Neuroscience*, den *Matthias H. Hennig* entwickelt hat.

CTAN:biblio/bibtex/contrib/
jneurosci

gnuplottex von *Lars Kotthoff* dient zum Einbinden von *gnuplot*-Grafiken in LaTeX-Dokumenten.

CTAN:macros/latex/contrib/
gnuplottex

MemoirChapStyles demonstriert, wie man Überschriften bei Verwendung von *memoir* gestalten kann. Autor ist *Lars Madsen*.

CTAN:info/MemoirChapStyles

showexpl ermöglicht es, den LaTeX-Quellcode und die daraus sich ergebende LaTeX-Ausgabe in demselben Dokument zu setzen. Autor ist *Rolf Niepraschk*.

CTAN:macros/latex/contrib/
showexpl

xifthen von *Josselin Noirel* erweitert die Funktionen des Standardpakets *ifthen*. Der Befehl `\ifthenelse` wird so neugefasst, dass er prüft, ob sein erstes Argument leer ist oder nicht. Auch kann man auf komplexe Ausdrücke zugreifen, die von dem Paket *calc* eingeführt werden. Erfordert ϵ -TeX.

CTAN:macros/latex/contrib/
xifthen

abc bietet ein Interface für LaTeX zur *ABC-Plus*-Notation für Musik (vgl. <http://abcplus.sourceforge.net>). Autor ist *Enrico Gregorio*.

CTAN:macros/latex/contrib/abc

arial nun auch für L^AT_EX (genauer: *URW A030*). Autor ist *Walter Schmidt*.

CTAN:fonts/urw/arial

randtext von *Charles Duan* dient zum Verschleiern von Text, insbesondere von Emailadressen, die bei Online-Veröffentlichungen vor dem automatisierten Zugriff durch Spammer geschützt werden sollen.

CTAN:macros/latex/contrib/
randtext

stellenbosch ist eine L^AT_EX-Klasse der Universität Stellenbosch in Südafrika. Autor ist *Danie Els*.

CTAN:macros/latex/contrib/
stellenbosch

linearA von *Apostolos Syropoulos* bietet ein Paket zur Wiedergabe von Texten, die in der *Linear A*-Schrift geschrieben wurden.

CTAN:fonts/linearA

classicthesis ist ein neues Paket zum Schreiben von wissenschaftlichen Arbeiten. Autor ist *André Miede*.

CTAN:macros/latex/contrib/
classicthesis

cellspace von *Josselin Noirel* dient zum Definieren des Abstands von Zelleninhalt in Tabellen zu den `\hlines` (also dem sogenannten »Padding«).

CTAN:macros/latex/contrib/
cellspace

blockdraw_mp stellt METAPOST-Makros zum Zeichnen von Blockdiagrammen und *bond graphs* zur Verfügung. Autor ist *Henrik Tidefelt*.

CTAN:graphics/metapost/
contrib/macros/blockdraw_mp

icsv für Beiträge zur ICSV-Konferenz. Autor ist *Will Robertson*.

CTAN:macros/latex/contrib/
conferences/icsv

pracjournal ist die L^AT_EX-Klasse für die Online-Zeitschrift *PracT_EX Journal*, die man unter <http://www.tug.org/pracjournal> findet. Autor ist *Will Robertson*.

CTAN:macros/latex/contrib/
pracjournal

moderncv von *Xavier Danaux* ist eine neue Klasse, mit der man seinen Lebenslauf für eine Bewerbung setzen kann.

CTAN:macros/latex/contrib/
moderncv

LingTrees dient zum Formatieren von Baumstrukturen, wie sie in der Linguistik Verwendung finden. Autor ist *Avery D. Andrews*.

CTAN:macros/latex/contrib/
lingtrees

eskd erzeugt ein typisches Layout für Dokumente in russischer Sprache. Autor ist *Konstantin Korikov*.

CTAN:macros/latex/contrib/
eskd

visualFAQ ist eine PDF-Datei, die anhand eines Musterdokuments mit *lorem ipsum*-Blindtext typische häufig auftretende Probleme visuell demonstriert und gleichzeitig zu den Antworten in der englischen UK T_EX-FAQ von *Robin Fairbairns* verlinkt (setzt *Adobe Reader 6* oder *7* voraus). Der Ansatz ist hilfreich, wenn man Schwierigkeiten hat, sein Problem in Worte zu fassen, weil man nicht weiß, wie die Fachausdrücke dazu heißen. Autor ist *Scott Pakin*.

cooltooltips ist das Paket, mit dem die PDF-Hyperlinks in *visualFAQ* erstellt worden waren: Wenn man mit der Maus darüber fährt, öffnet sich ein Pop-up-Fenster mit Hilfeinhalt.

CTAN:info/visualFAQ

CTAN:macros/latex/contrib/
cooltooltips

coverage versieht wissenschaftliche Aufsätze mit einer Titelseite, die auch die BibT_EX-Daten der Veröffentlichung und einen urheberrechtlichen Vermerk einschließen. Daten zu Autor und

Institut können in einer Konfigurationsdatei abgelegt werden. Autor ist *Matthias Mühlich*.

CTAN:macros/latex/contrib/
coverage

lsc von *Bernd Westphal* dient zum Setzen von »*Live Sequence Charts*« nach *Damm* und *Harel*.

CTAN:macros/latex/contrib/lsc

cmll stellt Symbole aus der linearen Logik für die Schriftarten *Computer Modern* und *AMS Euler* zur Verfügung. Autor ist *Emmanuel Beffara*.

CTAN:fonts/cmll

prerex sind ein Programm in C und eine *sty*-Datei zum interaktiven Erstellen von »*prerequisite charts*«, wie sie im nordamerikanischen Bildungswesen und in der Informatik gebräuchlich sind (Netzpläne). Autor ist *Bob Tennent*.

CTAN:graphics/prerex

mem ist experimenteller Code (*pre-alpha*), der im Rahmen des Projekts *Lamed* (»*L^AT_EX* + *Aleph*«; vgl.: <http://mem-latex.sourceforge.net>) entstanden ist. Autor ist *Javier Bezos*.

CTAN:macros/latex/exptl/mem

wordlike ist ein einfaches LaT_EX-Paket, das das Layout eines Dokuments dem anpaßt, was man von

einer Textverarbeitung kennt.
Autor ist *Jürgen Fenn*.

CTAN:macros/latex/contrib/
wordlike

exceltex ist ein kleines L^AT_EX-Paket von *Hans-Peter Doerr*, mit dem man auf Excel-Dateien zugreifen kann. Beigefügt ist ein Hilfsprogramm in Perl, das die Übernahme von Tabellen und einzelnen Zellen aus Excel-Dateien in L^AT_EX-Dokumente vereinfacht.

CTAN:macros/latex/contrib/
exceltex

xytree von *Koanghi Un* dient zum Zeichnen von Baumstrukturen, wie sie in der Linguistik gebräuchlich sind.

CTAN:macros/latex/contrib/
xytree

Galician-hyphenation sind erstmals galizische Trennmuster. Autor ist *Javier Múgica de Rivera*.

CTAN:language/hyphenation/
glhyph.tex

visualtex ist ein neuer Texteditor für Windows, den der Entwickler *Alex Romanenko* mit *MuItEdit* und *WinEdt* vergleicht.

CTAN:systems/win32/visualtex

telprint dient zum Setzen von Telefonnummern – bekannt aus *de.comp.text.tex*. Autor ist *Heiko Oberdiek*.

CTAN:macros/latex/contrib/
oberdiek/telprint.sty

malayalam-omega von *Alex A.J.* bietet eine Omega-Unterstützung für die indische Sprache Malayalam.

CTAN:macros/omega/latex/
contrib/malayalam-omega

sarabian archaic ist eine Schrift, die es bisher nur in METAFONT gab. Neu sind die nunmehr verfügbaren Type1-Fonts. Sie wurden einer alten Handschrift nachempfunden, die im südlichen Arabien von etwa 600 v. Chr. bis ca. 1600 n. Chr. in Gebrauch war. Autor ist *Peter Wilson*.

CTAN:fonts/archaic

active-conf ist eine Dokumentenklasse zu der gleichnamigen diesjährigen Tagung in Adelaide. Autor ist *Will Robertson*.

CTAN:macros/latex/contrib/
conferences/active-conf

toptesi dient zum Setzen wissenschaftlicher Arbeiten nach den Vorgaben, die an italienischen Universitäten gelten. Autor ist *Claudio Beccari*.

CTAN:macros/latex/contrib/
toptesi

L^AT_EX Beta-Test ist ein neuer L^AT_EX-Release, den das *L^AT_EX-Team* am 7. Februar 2006 zum

öffentlichen Beta-Test freigegeben hat. Unter <http://www.informatik.uni-freiburg.de/~oberdiek/tmp/latex-tds/> stellt *Heiko Oberdiek* ein ZIP-Archiv bereit, das man zur Installation einfach in den `texmf`-Baum entpacken kann.

CTAN:macros/latex/beta-test

RAT_EX bietet eine Sammlung von Paketen für die praktische Arbeit im RechtsAnwaltsbüro. Es soll demnächst auch auf CTAN zu *juRAT_EX* umbenannt werden. Die Entwicklerversion findet man unter <http://homepage.mac.com/vatolin/juRATeX/FileSharing21.html>. Autor ist *Friedrich Vosberg*.

CTAN:macros/latex/contrib/
rtklage

marginnote von *Markus Kohm* stellt den Befehl `\marginnote` zur Verfügung, der überall dort verwendet werden kann, wo `\marginpar` nicht funktioniert (z. B. in Gleitobjekten, in Fußnoten oder in Rahmen, die mit dem Paket *framed* erzeugt wurden).²

CTAN:macros/latex/contrib/
marginnote

mab2bib sind zwei nützliche Python-Skripte von *Henning Hraban*

Ramm, mit denen man bibliographische Angaben aus allgemeinen Bibliothekskatalogen für die Verwendung mit `BIBTEX` konvertieren kann: *mab2bib* konvertiert Datensätze im MAB-Standard zu `BIBTEX`.³ *utf2latex* wandelt zudem `BibTeX`-Daten von `utf-8` nach ASCII.⁴

CTAN:biblio/bibtex/utis/
mab2bib

dinbrief-gui ist ein flinkes Tool (`exe`-Datei und `Tcl/Tk`-Skript), das eine Schnittstelle zur Verwendung der Briefklasse *dinbrief* bereitstellt. Autor ist *Wolf-Dieter Busch*.

CTAN:support/dinbrief-gui

gentabtex von *Manuel Gutierrez Algaba* ist ein Python-Backend zur Erzeugung von Tabellen in `LATEX`.

CTAN:support/gentabtex

emerald enthält die `LATEX`-Unterstützung für einige ECF-Fonts (»*emerald city fontworks*«, <http://www.speakeasy.org/~ecf/>), einschließlich der `Type1`-Fonts selbst. Autor ist *M. Cesar Rodriguez*.

CTAN:fonts/emerald

²Siehe auch Seite 15 in diesem Heft.

³Vgl. <http://www.ub.uni-konstanz.de/bibtex.htm>.

⁴Vgl. <http://www.digibib.net>.

MFwL ist eine Präsentation (»*Making Friends with L^AT_EX*«), die *Rohit Vishal Kumar* entwickelt hat.

CTAN:info/MFwL

dirtree von *Jean-Côme Charpentier* stellt Datei-Verzeichnisse als Baumansicht dar, die derjenigen des *Windows Explorers* ähnelt.

CTAN:macros/generic/dirtree

duerer-latex sowie *pandora-latex* und *punk-latex* sind einige L^AT_EX-Makros von *Palle Jörgensen* zur Verwendung der jeweiligen Fonts, die bereits in T_EXLive 2005 enthalten waren.

CTAN:macros/latex/contrib/
duerer-latex

CTAN:macros/latex/contrib/
pandora-latex

CTAN:macros/latex/contrib/
punk-latex

Winfonts v1.0 bietet eine Unterstützung für die Standardschriftarten, die üblicherweise unter Win-

dows verwendet werden und die zusammen mit Windows XP im Format *Type 1* ausgeliefert werden. Autor ist *Paul Pichaureau*.

CTAN:fonts/winfonts

MPStoEPS ist ein Perl-Skript, um die Ausgabe von METAPOST (MPS) in *Encapsulated PostScript* (EPS) umzuwandeln. Alle Fonts werden eingebettet. Autor ist *Troy Henderson*.

CTAN:graphics/metapost/
contrib/tools/mpstoeps

gapfill (»*Graphics as PostScript Filtered for L^AT_EX*«) von *Herbert Möller* soll den Umgang mit Zeichnungen unter L^AT_EX vereinfachen. Die Zeichnung kann aus einem Zeichenprogramm als PostScript-Datei exportiert werden. Mithilfe eines Perl-Skripts kann diese dann in eine *picture*-Umgebung umgesetzt werden (vgl. *ders.*, DTK 1/2006, S. 25 ff.).

CTAN:graphics/gapfill

Rezensionen



Foto: Lehmanns Buchhandlung

Matthias Kalle Dalheimer: $\text{\LaTeX}$ – kurz & gut

Uwe Siart und Herbert Voß

Im Rahmen der allgemeinen $\text{\LaTeX}$ -Renaissance gibt es nach langer Zeit inzwischen auch eine zweite Auflage von *$\text{\LaTeX}$ – kurz & gut*, einer thematisch gut gegliederten $\text{\LaTeX}$ -Befehlsreferenz mit Kurzerklärungen und Hinweisen für den etwas erfahreneren $\text{\LaTeX}$ -Anwender aus der Reihe *O'Reillys Taschenbibliothek*. Sie beschränkt sich allerdings bewusst auf den Grundbefehlssatz des $\text{\LaTeX}$ -Formats. Auf wichtige Erweiterungen, die inzwischen verpflichtend in jeder $\text{\LaTeX}$ -Distribution enthalten sind, geht sie nicht ein. Dies ist teilweise verständlich, denn eine bewusst kurz gehaltene Referenz muss auswählen. Dennoch hätte man im Abschnitt über mathematische Formeln in einer aktuellen Ausgabe unbedingt die Ergänzungen durch das zum Standard gehörende `amsmath`-Paket erwartet. Der Abschnitt über das Grafikpaket `PSTricks` ist sehr knapp ausgefallen. Der Funktionsumfang des Basispakets ist nur zu einem kleinen Teil enthalten und die dort angeführten Befehle zur Farbdefinition haben in einer Auflage des Jahres 2005 nichts mehr zu suchen.

Grundsätzlich ist *$\text{\LaTeX}$ – kurz & gut* ein handlicher und übersichtlicher Katalog grundlegender $\text{\LaTeX}$ -Befehle, dem man aber an einigen Stellen sein Entstehungsjahr 1998 noch deutlich anmerkt. Die vielen kleinen Unzulänglichkeiten werten den Vorteil dieser sehr kurzen Befehlsreferenz deutlich ab. Dies ist bedauerlich, denn eine derartige Veröffentlichung hat in jedem Fall ihre Berechtigung. Schließlich will nicht jeder immer ein Buch von mehreren hundert Seiten durchblättern müssen, wenn es nur um die Syntax eines Befehls geht. Die zusätzliche Lektüre von aktuellen Auflagen der vom Autor empfohlenen weiterführenden Literatur ist auf jeden Fall anzuraten.



Kalle Dalheimer: *$\text{\LaTeX}$ – kurz & gut*; 2. Aufl. Oktober 2004; ISBN 3-89721-500-4; 78 Seiten; 8,00 € bzw. 13,90 SFR

»Der L^AT_EX-Begleiter« von Frank Mittelbach, Michel Goossens u. a.

Jürgen Fenn

Frank Mittelbach, Michel Goossens und weitere Mitarbeiter des L^AT_EX-Projekts haben eine neue, zweite Auflage ihres »L^AT_EX-Begleiters« vorgelegt, die nunmehr auch in einer deutschen Ausgabe erschienen ist, die im folgenden vorgestellt wird.



Zehn Jahre später

Seither [seit 1994] ist einiges an Zeit vergangen. L^AT_EX hat sich ein wenig verändert, die L^AT_EX-Landschaft erheblich.

Markus Kohm und *Jens-Uwe Morawski* [8, S. 14]
über die Entwicklung von KOMA-Script

Erich Ruff [13] hat den langen Weg der Entwicklung von L^AT_EX zu dem, was es heute ist, vor kurzem in dieser Zeitschrift nachvollzogen. Viele Ergänzungspakete, Schriftarten, Konverter und weitere Zusatzprogramme sind in dieser Zeit hinzugekommen, die das Arbeiten mit L^AT_EX ganz wesentlich erleichtern. Die Folge war, dass man der ersten Auflage des L^AT_EX-Begleiters [5], [6] ihr Alter durchaus angemerkt hat. Es ist gleichwohl eine zuverlässige Referenz, von der d. Verf. bei der Arbeit immer profitiert hat. Umso dankbarer ist man den Autoren dafür, dass sie ihr Werk zehn Jahre nach Erscheinen der ersten englischen Auflage [5] überarbeitet haben und dass der Verlag nunmehr auch die zweite Auflage in einer deutschen Übersetzung herausgebracht hat.

Der erfahrene Anwender wird sich nun – nicht zuletzt mit Blick auf den Preis – fragen, ob er dieses »Update« wirklich brauche oder ob ihm die erste Auflage zusammen mit der auf seinem System ohnehin vorhandenen Dokumentation genüge? Tatsächlich ist fraglich, ob ein System, das so umfangreich dokumentiert ist wie L^AT_EX – die Verzeichnisse `/texmf/doc` und `/localtexmf/doc` der MiK_TE_X-Distribution d. Verf. haben derzeit einen Umfang von zusammen immerhin 275 MB –, noch einer Ergänzung in Form eines Buches mit einem Umfang von über 1100 Seiten bedarf oder ob es sich dabei vielmehr um eine Art »*Overkill*« handelt. Der Anfänger könnte demgegenüber angesichts dieses Umfangs womöglich eingeschüchtert abwinken – das könne er niemals in vertretbarer Zeit lesen, schon gar nicht neben dem Studium oder neben dem Beruf.

Für wen eignet sich dieses Buch also? Um es vorwegzunehmen: Es eignet sich für beide, für den fortgeschrittenen L^AT_EX-Anwender ebenso wie für den Anfänger – soweit er es gerne »etwas genauer« wissen möchte und vorausgesetzt, er will sich ganz auf L^AT_EX einlassen und der Office-Textverarbeitung dauerhaft »Lebwohl« sagen. Wer sich in dieser Typisierung wiederfindet, dem soll der nachfolgende Text eine Hilfestellung dazu geben, um herauszufinden, ob er sich den neuen *Begleiter* zulegen sollte. Nicht verschwiegen werden soll, dass es bereits ein paar ganz überwiegend wohlwollende, aber recht knapp gefasste Besprechungen zu der englischen Ausgabe der Neuauflage [1],[3] und zur deutschen Übersetzung [4] gegeben hat.

Äußerlichkeiten

Der erste Eindruck, den das Buch vermittelt, ist schon recht positiv: Man hält ein stabil und schön gebundenes Buch in Händen, auf dem Titel prangt, im Gegensatz zur englischen Ausgabe, die einen Kompass zeigt, wie von der ersten Auflage her bekannt, ein beruhigend dreinblickender Bernhardiner, diesmal vor dem Matterhorn platziert. Ein blaues Lesebändchen ragt aus dem Buchblock hervor. Beim ersten Durchblättern fallen die Druckqualität und das gute Papier auf: Ein strapazierfähiger Band, dem man zutraut, auf Jahre hinaus als Handbuch »durchzuhalten«.

Selbstverständlich wurde das Buch selbst mit L^AT_EX gesetzt. Die »Entstehungsgeschichte« des Layouts wird im Anhang beschrieben (S. 1135 ff.) Der Druck ist durchgehend zweifarbig ausgeführt: Wichtiger Code, aber auch Überschriften und Marginalien sind in blauer Farbe hervorgehoben, was die Lesbarkeit fördert und das Layout insgesamt lebendiger macht, möglicherwei-

se aber auch das Erstellen von Fotokopien auf manchen Kopierern, die gegenüber der Farbe Blau »blind« sind, erschweren könnte. Etwas irritierend, weil zunächst ungewohnt, aber beim Lesen und Kopieren mitunter auch unpraktisch erscheint der etwas schmal geratene innere Seitenrand, in dem die vielen Beispiele durchnummeriert sind, deren Quellcode man auf der beiliegenden CD-ROM wiederfindet. Ebenfalls auf der CD findet man eine teilweise aktualisierte T_EXLive-Distribution aus dem Jahr 2003 für MS Windows, Linux und Mac OS X, die auch direkt von der CD ausgeführt werden kann.

Der sehr sorgfältigen deutschen Übersetzung liegt der zweite korrigierte Nachdruck der englischen Ausgabe zugrunde. Es handelt sich ganz sicherlich um das am besten »lesbare« Buch unter den d. Verf. bekannten L^AT_EX-Einführungen, eine so gepflegte Sprache wie im *L^AT_EX-Begleiter* sucht man in der EDV-Literatur leider meist vergebens. Das Buch ist auch insoweit vorbildlich. Eine Übersicht über die in der deutschen Ausgabe gefundenen Fehler findet man übrigens online unter <http://www.latex-project.org/guides/lb2.err>.

Vorbereitungen – und der Ansatz der Autoren

Elementare L^AT_EX-Kenntnisse werden beim Leser vorausgesetzt. Idealerweise sollte das (derzeit leider nicht über den Buchhandel lieferbare) L^AT_EX-Handbuch von *Leslie Lamport* [10] bereits durchgearbeitet (und verstanden) worden sein, es dürfte aber auch jede andere gute Einführung genügen. Der Leser sollte bereits mit der Arbeitsweise bei Verwendung von L^AT_EX vertraut sein, wenn er sich dem *Begleiter* zuwendet, denn der *Begleiter* setzt dort ein, wo die – in diesem Sinne »vorbereitenden« – Titel aufhören und vertieft alle einschlägigen Themenkreise, die man aus der übrigen Literatur kennt, also beispielsweise das Arbeiten mit mehreren Sprachen in einem Dokument oder die Erstellung von Literaturverzeichnissen sehr grundlegend, um sich dann einem äußerst souveränen Umgang mit diesen Themen zu widmen. So erhält man nicht nur eine Anleitung zu L^AT_EX, sondern gleichzeitig eine Einführung in die EDV-gestützte Arbeit mit Texten und in die Gestaltung von Drucksachen, die das Verständnis für den »richtigen« Umgang mit L^AT_EX fördert.

Die Fortsetzung des Berichts

[...] a 90% rewrite of all existing material (ie nothing much left) and about an additional 500 new pages [...]

Frank Mittelbach [11]

Darüber hinaus handelt es sich um eine Art »Werkstattbericht« aus dem Herzen des L^AT_EX-Projekts (Projekthomepage: <http://www.latex-project.org/>), denn die Autoren sind seit langem diejenigen Akteure, die die Entwicklung von L^AT_EX maßgeblich tragen. Schon die erste Auflage des Buches beschrieb ausführlich die Verwendung von Ergänzungspaketen, die letztlich eine Stärke von L^AT_EX darstellen, weil sie das System in hohem Maße flexibel machen und halten.¹ Die erneute Auseinandersetzung mit den insoweit verfügbaren Standardlösungen hat aber über die bloße Neu-Beschreibung der Pakete in vielen Fällen zu deren Überarbeitung beigetragen, was zu einer Welle an Updates in den Jahren 2002 und 2003 kurz vor Erscheinen der englischen Neuauflage geführt hatte.

Die Unterschiede zur ersten Auflage sind erheblich angesichts einer beinahe Verdoppelung des Buchumfangs mit einem Zuwachs um mehrere hundert Seiten und einer Neubearbeitung, die so gründlich und umfassend erfolgt ist, dass man tatsächlich sagen kann, man halte ein neues Buch in Händen. Der grundlegende Aufbau des *Begleiters* wurde zwar beibehalten, der Text ist aber im Ganzen viel ausführlicher gefaßt als zuvor. Man hat den Eindruck, die Autoren hätten sich keinen bedeutenden äußeren Zwängen mehr unterordnen müssen, dass insbesondere der Platz, der ihnen zur Verfügung stand, keine wesentliche Grenze mehr war. So kommt es, dass etwa die Einführung in die Erstellung von Literaturverzeichnissen im Vergleich zur ersten Auflage [6, S. 385–438] noch einmal um 84 Seiten vermehrt wurde und sich nun über zwei Kapitel erstreckt. Das ist der Darstellung durchweg sehr zugute gekommen.

Die *Einleitung* stellt somit weiterhin das L^AT_EX-System im Ganzen vor, nunmehr allerdings erweitert um einen historischen Rückblick, der mehr ist als

¹Umgekehrt könnte man sagen, dass die ungeheure Anzahl von mittlerweile verfügbaren Paketen gleichzeitig ein Hindernis für den Anwender darstellen kann, weil sie die Übersichtlichkeit des Systems als ganzes verschlechtern. Kaum jemand blickt noch durch, auch die Bearbeiter des *T_EX Catalogue* (vgl. <http://texcatalogue.sarovar.org/>) kämpfen mitunter mit der Datenflut. Aber das ist ein anderes Thema.

bloße Nostalgie, weil er das Verständnis für die Funktion des Systems und die Technik des Arbeitens mit L^AT_EX schärft (16 Seiten). Der *Begleiter* vertieft hier elementare Einführungen, indem der Aufbau von L^AT_EX-Dokumenten erläutert wird (7 Seiten). Sodann widmen sich die Autoren ausführlich der Formatierung von Dokumenten (Überschriften, Verzeichnisse, Fuß- und Endnoten, Listen und wörtliche Zitate, Seitenlayout, Tabellen und Gleitobjekte, auf insgesamt 313 Seiten – das Kapitel kann abgerufen werden unter: <http://www.latex-project.org/guides/lb2-ch4.pdf>), um sich dann dem Thema *Zeichensätze und Kodierungen* zuzuwenden, das auch viele routinierte Anwender und Entwickler schon öfter in Verlegenheit gebracht haben dürfte (140 Seiten). Wem der mathematische Formelsatz also wichtig ist, sollte auf jeden Fall einen Blick in die Neuauflage werfen, denn das neue Kapitel zu $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX (78 Seiten) ist nunmehr auf dem aktuellen Stand. Ebensoviele fehlen eine Darstellung des Pakets `babel` (auf 56 Seiten) und Einzelheiten zum Umgang mit Grafiken (ebenfalls 56 Seiten). Weiter geht es mit Kapiteln über die Erstellung von Indizes und das Zitieren von Fundstellen sowie die *Erzeugung von Literaturverzeichnissen* für wissenschaftliche Arbeiten (zusammen 178 Seiten). Abgerundet wird das Buch durch umfangreiche Hinweise zum Entwickeln und Testen eigener Pakete, einschließlich bisher überwiegend unveröffentlichter Erläuterungen (so: [1, S. 276]) über die mitunter mysteriösen Fehlermeldungen von T_EX: Das Kapitel 14 und die Anhänge A und B machen immerhin 150 Seiten aus.

Aus den Seitenangaben ist bereits ersichtlich, dass es sich wahrlich nicht um dünne Bretter handeln kann, die hier gebohrt werden. Die Lektüre erfordert ein gewisses Maß an aktiver Mitarbeit beim Leser und die Bereitschaft, sich auf einen längeren Gedankengang und auf die Erörterung grundlegender Zusammenhänge einzulassen. Gleichzeitig ist das Buch aber auch zum selektiven Lesen gut geeignet. Das sehr ausführliche und übersichtlich gestaltete Stichwortverzeichnis lädt mit einem Umfang von 93 Seiten zum zielgenauen Nachschlagen ein. Auch das Literaturverzeichnis des *Begleiters* hat es »in sich«: Auf 22 Seiten bietet es eine kommentierte und vollständige Bibliographie zu L^AT_EX – von der Paketanleitung auf CTAN bis hin zu wissenschaftlichen Fundstellen (Bibliographie und Index sind online verfügbar unter: <http://www.latex-project.org/guides/lb2-ap4.pdf>).

Ein Beispiel

Der *Begleiter* ist eine Fundgrube an praktischem Wissen zum Umgang mit L^AT_EX. Die Autoren fallen aber nicht »mit der Tür ins Haus«, sondern führen den Leser fundiert und umsichtig von den grundlegenden Problemen herkommend zu den Lösungen hin, der Band ist voll von oft eleganten Codebeispielen.

Die Darstellung zum Umgang mit wissenschaftlichen Quellen und zur Erstellung des Literaturverzeichnisses soll das verdeutlichen (S. 707–844): Man erhält zunächst einen Überblick zu den grundlegenden Schemata, die zum Zitieren von Literatur in wissenschaftlichen Arbeiten üblich sind: Das »*Kurztitelschema*«, das »*Autor-Jahr-Schema*« und das »*numerische Schema*«. Dabei wird deutlich, dass es nicht *ein* Modell für die Angabe von Fundstellen gibt, sondern dass sich in den verschiedenen Fächern je eigene »Zitierkulturen« eingebürgert haben. Für die dabei sich ergebenden Bedürfnisse und Probleme gibt es jeweils Lösungen in L^AT_EX, man kann seine Bibliographie entweder in einer **thebibliography**-Umgebung händisch erstellen, oder man kann diese Aufgabe Bib_TE_X übertragen; in diesem Fall ist eine externe Literaturdatenbank zu erstellen, eine solche wird beispielhaft gezeigt, und es sind mehrere latex- und bibtex-Läufe in einer bestimmten Reihenfolge zu starten. Der hierbei sich ergebende Datenfluss wird graphisch veranschaulicht. Der eigentliche Zugriff auf die bibliographischen Daten erfolgt schließlich durch den **cite**-Befehl, der die Fundstelle im laufenden Text ausgibt. Wie man auf dieser Grundlage praktisch arbeitet, und welche Möglichkeiten (aber auch: welche Grenzen) es gibt, Fundstellen und Literaturverzeichnis zu formatieren, wird sodann ausführlich für jedes der drei eingangs genannten Schemata erläutert. Dabei werden die Standardlösungen **natbib** und **jurabib** ausführlich vorgestellt – letzteres so ausführlich, dass auch Sozial- und Geisteswissenschaftler, die sich zunehmend L^AT_EX zuwenden, voll auf ihre Kosten kommen werden (der *Begleiter* ist nämlich auch eines der wenigen Bücher, denen man wirklich nicht nachsagen könnte, sie wären allzu mathematiklastig geraten). Ein Überblick zur Erstellung mehrerer Bibliographien in einem Dokument schließt sich an. Es folgt die Darstellung des Datenbankformats für Bib_TE_X und seine Epigonen, sowie ein aktueller Überblick über die derzeit verfügbaren Tools zum Bearbeiten und zum Konvertieren von Bib_TE_X-Dateien, einschließlich der graphischen Frontends **Jabref** und **Pybliographer**. Den Abschluss bildet ein Überblick zur Erweiterung bestehender oder gar zur Entwicklung eigener Bib_TE_X-Stile. Wer nach alledem noch offene Fragen hat, dem ist wahrscheinlich wirklich nicht mehr zu helfen.

Vorzüge

Gegenüber anderen »großen« L^AT_EX-Büchern stechen vor allem zwei Kapitel hervor: Die Behandlung mehrsprachiger Dokumente mithilfe von **babel** und die Darstellung des Themas *Zeichensätze und Kodierungen*. Hier haben sich – rückblickend – ganz sicherlich zugleich die größten Fortschritte ergeben, insbesondere durch die Einführung von *Unicode* und durch die fortwährende Entwicklung immer neuer Fonts, die L^AT_EX zu einem System gemacht haben, das sich längst nicht nur für den mathematischen Formelsatz empfiehlt. In diesem Zusammenhang sei auch erwähnt, dass die Autoren dem neuen Index-Tool **xindy**, einer für nicht-englische Sprachen besonders geeigneten Alternative zu **makeindex**, einen eigenen Abschnitt widmen.

Das Kapitel zu den Zusammenhängen zwischen den in L^AT_EX verfügbaren Zeichensätzen und den Kodierungen gehört zur Pflichtlektüre für jeden, der sich mit diesen Fragen ernsthaft beschäftigen möchte. Es gibt hierzu schlicht nichts Vergleichbares in der übrigen Literatur. Das Thema ist alles andere als trivial, und kaum ein anderer Autor hat sich dem bisher überhaupt gewidmet. Fündig wird hier, wer »schon immer einmal« mehr wissen wollte zum *Einbinden neuer Zeichensätze* oder zum *Setzen einzelner Schriftattribute*. Der Zusammenhang zwischen Eingabe- und Font-Kodierungen wird ausführlich erklärt, einschließlich der L^AT_EX-internen Zeichendarstellung LICR. Auf einer für den durchschnittlichen Anwender sicherlich vertrauteren Ebene werden im Übrigen aber auch der Zugriff auf PostScript-Schriften mittels **PSNFSS** oder die vielen heute gebräuchlichen Symbol-Fonts gezeigt.

Etwas fehlt. . .

Ein Makel, den kaum ein Buch abstreifen kann, das sich mit Informatik oder Computern beschäftigt, ist die Tatsache, dass man so viele interessante Dinge *nicht* behandeln kann.

Henning Behme und Stefan Mintert
[2, zu Anfang von Abschnitt 2.1]

Kann auf 1100 Seiten zum Thema L^AT_EX noch etwas »fehlen«? Nicht viel, aber doch einiges, was zumindest insoweit nicht allzu überraschend ist, als ja auch diese Rezension nur einen Ausschnitt aus dem umfangreichen Buch behandeln kann. Hier zeigt sich erst, wie »mächtig« das Thema L^AT_EX tatsächlich (geworden) ist. Die – berechnete – Lobrede, die hier geführt wird,

soll deshalb an dieser Stelle unterbrochen werden, um aufzuzeigen, welche Themen von den Autoren nicht behandelt werden und auf welche Quellen der Leser insoweit zurückgreifen kann.

Völlig unerwähnt bleiben die zahlreichen »Office«-Anwendungen, die es mittlerweile gibt. Insbesondere fehlt jeder Hinweis darauf, dass man mit L^AT_EX auch sehr gut und leicht Briefe schreiben kann. Das ist wirklich schade, denn die mittlerweile verfügbaren Briefklassen, vor allem `scr1ttr2` aus dem KOMA-Script-Paket und `dinbrief`, machen sogar das Briefeschreiben zu einer sehr einfachen Übung. Gerade angesichts der heute wirklich nicht mehr nachvollziehbaren und angesichts der in vorgenannten ganz vorzüglichen Paketen obsolet gewordenen »Klimmzüge« zur Anpassung von `letter.cls` im ersten Band der *Kopkaschen* L^AT_EX-Trilogie [9, S. 277 ff.] wäre es angezeigt gewesen, auf dieses Thema einzugehen. Man wird deshalb auf die diesbezüglichen Paketdokumentationen zurückgreifen müssen.

Die wichtigen Ergänzungen für die Standardklassen KOMA-Script und Memoir werden nur kurz erwähnt (S. 244 f.), was allerdings vertretbar erscheint, weil den Paketen umfangreiche eigene Anleitungen beigelegt sind [8], [15].

Ebenso fehlen Ausführungen zu den vielen gebräuchlichen Konvertern, die es mittlerweile für L^AT_EX-Dateien gibt. Man denke an `tex4ht`, `latex2html` oder `latex2rtf`. Dies kann indessen auf die »Arbeitsteilung« zwischen den Bänden der *Begleiter*-Reihe zurückgehen, in der ein eigener Band hierfür vorgesehen war [7], für den sich bisher leider keine Neuauflage abzeichnet. »L^AT_EX. Das Praxisbuch« [12] widmet dem Thema ein eigenes Kapitel, dessen Abschnitt zu dem neuen, sehr mächtigen Konverter `writer2latex` (für die Richtung `OpenOffice.org` → L^AT_EX 2_ε) auch online verfügbar ist unter http://www.hj-gym.dk/~hj/writer2latex/auszug_w2l.pdf.

Das sehr umfangreiche und mächtige Paket `PSTricks` wird leider auch nur sehr am Rande erwähnt. Hierfür stehen aber umfangreiche Einführungen bei [14] und [12] zur Verfügung.

Auch zu Texteditoren und speziellen Entwicklungsumgebungen sowie zur Arbeit mit `makefiles` zur Erstellung komplexer Dokumente findet man im *Begleiter* leider gar nichts.

Dies alles mindert den Wert des Buches jedoch in keiner Weise.

Ausblick. . .

It is kind of my homage to L^AT_EX 2_ε and its contributors before I finally turn my back on it and return to L₃ in earnest again.

Frank Mittelbach [11]

So bleibt zu hoffen, dass die Arbeit des L^AT_EX-Projekts nun, da der neue *Begleiter* vorliegt, wie von *Frank Mittelbach* [11] erwähnt, voranschreiten möge und L^AT_EX bald noch weiter verbessert wird.

Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle, Chris Rowley, Christine Detig und Joachim Schrod, »Der L^AT_EX-Begleiter«, zweite, überarbeitete und erweiterte Auflage, Übersetzung von Claudia Krysztofiak und Rebecca Stiels, München 2005, Pearson Studium, 1138 Seiten, ISBN 3-8273-7166-X, mit einer CD-ROM, 59,95 Euro (in Österreich: 61,70 Euro).

Literatur

- [1] Claudio Beccari: *Book review: The L^AT_EX Companion, Second Edition; TUGboat*; 24(2), S. 275–277; 2003; Online verfügbar unter: <http://www.tug.org/TUGboat/Articles/tb24-2/tb77becc.pdf>.
- [2] Henning Behme und Stefan Mintert: *XML in der Praxis*; 1998–2004; Onlineresource: <http://www.linkwerk.com/pub/xmlidp/2000/wp-history.html>.
- [3] Javier Bezos: *The L^AT_EX Companion, segunda edición; T_EXemplares*; 6(7), S. 21–23; 2005; Online verfügbar unter: <http://filemon.mecanica.upm.es/CervanTeX/texemplares7.pdf>.
- [4] Jürgen Fenn: *Mittelbach et. al., Der L^AT_EX-Begleiter, 2. Aufl., 2005; c't*; 2/06, S. 212; 2006.
- [5] Michel Goossens, Frank Mittelbach und Alexander Samarin: *The L^AT_EX Companion; Tools and Techniques for Computer Typesetting*; 1994.
- [6] Michel Goossens, Frank Mittelbach und Alexander Samarin: *Der L^AT_EX-Begleiter*; 2000; korrigierter Nachdruck 2002.
- [7] Michel Goossens, Sebastian Rahtz et al.: *The L^AT_EX Web companion: integrating T_EX, HTML, and XML*; Tools and Techniques for Computer

- Typesetting; 1999; Deutsche Ausgabe u.d.T.: *Mit L^AT_EX ins Web. Elektronisches Publizieren mit T_EX, HTML und XML*; 2000.
- [8] Markus Kohm und Jens-Uwe Morawski: *KOMA-Script – Die Anleitung*; Aug. 2005; CTAN:/macros/latex/contrib/koma-script/scrguide.pdf.
 - [9] Helmut Kopka: *L^AT_EX*; Bd. 1; dritte Aufl.; 2000.
 - [10] Leslie Lamport: *L^AT_EX: A Document Preparation System: User's Guide and Reference Manual*; zweite Aufl.; 1994; korrigierter Nachdruck 1996. – Deutsche Ausgabe: *Das L^AT_EX-Handbuch*, 1995.
 - [11] Frank Mittelbach: *Posting in comp.text.tex*; 29. September 2003; Message-ID: bla6gm\$ulb\$3@online.de.
 - [12] Elke Niedermair und Michael Niedermair: *L^AT_EX. Das Praxisbuch*; Franzis' Professional Series; zweite vollständig aktualisierte und überarbeitete Neuauflage Aufl.; 2005.
 - [13] Erich Ruff: 2² – *Dante wird sechzehn*; DTK; 3/2005, S. 8 ff.
 - [14] Herbert Voß: *PSTricks – Grafik mit PostScript für T_EX und L^AT_EX*; Edition Dante; Dante e. V., Lehmanns Fachbuchhandlung; dritte, stark erweiterte und verbesserte Aufl.; 2006.
 - [15] Peter R. Wilson: *The Memoir Class for Configurable Typesetting – User Guide*; sechste Aufl.; Jan. 2004; CTAN:/macros/latex/contrib/memoir/memman.pdf.

Leserbriefe

Sehr geehrter Vorstand von DANTE e. V.,

ich halte mit Freude »die T_EXnische Komödie« Heft 2/2006 in den Händen. Besonders der zugehörige Tagungsband zur EuroT_EX 2005 war sehr erfreulich. Auch die beiden beigelegten gelben Aufkleber sind an sich eine gute Idee, wenn mir nicht ein Detail ihrer Beschriftung die gute Laune verdorben hätte. Wer ist denn auf die abwegige Idee gekommen, dort »professional typesetting with« drauf zu schreiben? Reicht es nicht, wenn man ständig auf der Straße oder in nahezu beliebigen Fernsehsendungen oder Zeitschriften von überflüssigen englischen Texten belästigt wird? Ich möchte eindringlich daran erinnern, dass das »D« in »DANTE« für deutschsprachig steht! Ich erwarte von meinem Lieblingsverein, dass er in solchen Fragen besonders sensibel ist.

Mit freundlichen Grüßen,

Euer treuer Leser Rolf Niepraschk

Antwort auf den Leserbrief

Da ich die Ausschreibung für den Wettbewerb der Aufkleber »verbrochen« habe, möchte ich auf den Leserbrief von Rolf Niepraschk antworten. Vor allem auch deshalb, weil ich in der Ausschreibung implizit eine englischsprachige Beschriftung vorgegeben habe. So stand im Text ein Vorschlag für den Aufkleber auf englisch: »formatted by T_EX«.

Hintergrund für die Idee des Aufklebers war es, einen Ersatz für die »Intel inside« bzw. »designed for Microsoft Windows« zu haben, die beide auch in englisch abgefasst sind – und, falls mir diese kleine Spitze erlaubt ist, letzteres auch ohne Englischkenntnisse zu benutzen ist, während man sich mit T_EX oder L^AT_EX und nur deutschen Sprachkenntnissen etwas schwerer tut.

Prinzipiell stimme ich Rolf Niepraschk zu, dass mittlerweile zu viele Anglizismen in der deutschen Sprache verwendet werden, vor allem dann, wenn man schon gar kein deutsches Wort mehr dafür findet, geschweige denn das englische nicht erklären kann. Aber diese Diskussion soll hier nicht geführt werden.

Meiner Meinung nach sind die Aufkleber sehr gelungen und erfüllen, zumindest bei mir, ihren Zweck, die Intel- und MS-Aufkleber zu ersetzen oder zu ergänzen. Und hier sind die englischen Begriffe auch passend.

Gruß,

Tobias Sterzl

Spielplan

Termine

21. 10. 2006 GuIT 2006
Pisa, Italien
<http://www.guit.sssup.it/>



9. 11.–11. 11. 2006 TUG 2006
Marrakesch, Marokko
<http://tug.org/tug2006/>

24.–25. 3. 2007 ConTeXt user meeting
Epen, Zuid-Limburg, Niederlande
<http://context.aanhet.net/epen2007/index.htm>

Stammtische

In verschiedenen Städten im Einzugsbereich von DANTE e.V. finden regelmäßig Treffen von T_EX-Anwendern statt, die für jeden offen sind. Im WWW gibt es aktuelle Informationen unter <http://www.dante.de/events/stammtische/>.

Aachen

Torsten Bronger
bronger@physik.rwth-aachen.de
Gaststätte Knossos
Templergraben 28
Zweiter Donnerstag im Monat, 19.00 Uhr

Berlin

Rolf Niepraschk
Tel.: 030/3 48 13 16
rolf.niepraschk@ptb.de
Gasthaus Pali-Eck
Koppenstr. 41
Zweiter Donnerstag im Monat, 19.00 Uhr

Bremen

Martin Schröder
Tel.: 04 21/2 23 94 25
martin@oneiros.de
Wechselnder Ort
Erster Donnerstag im Monat, 18.30 Uhr

Darmstadt

Karlheinz Geyer
geyerk.fv.tu@nds.tu-darmstadt.de
Restaurant Poseidon
Rheinstraße 41
64283 Darmstadt
Erster Freitag im Monat, ab 19.30 Uhr

Dresden

Carsten Vogel
lego@wh10.tu-dresden.de
Studentenwohnheim, Borsbergstraße 34,
Dresden, Ortsteil Striesen
ca. alle 8 Wochen, Donnerstag, 19.00 Uhr

Düsseldorf

Georg Verweyen
Georg.Verweyen@web.de
Bistro/Café Zicke
Bäckerstr. 5 a (Ecke Bergerallee)
40213 Düsseldorf
Zweiter Mittwoch in ungeraden Monaten,
20.00 Uhr

Erlangen

Walter Schmidt, Peter Seitz
w.a.schmidt@gmx.net
Gaststätte »Deutsches Haus«
Luitpoldstraße 25
Dritter Dienstag im Monat, 19.00 Uhr

Freiburg

Heiko Oberdiek
Tel.: 07 61/4 34 05
oberdiek@uni-freiburg.de
Wechselnder Ort
Dritter Donnerstag im Monat, 19.30 Uhr

Hamburg

Lothar Fröhling
lothar@thefroehlings.de
Wechselnder Ort
Vierter Dienstag im Monat, 19.30 Uhr

Hannover

Mark Heisterkamp
heisterkamp@rrzn.uni-hannover.de
Seminarraum RRZN
Schloßwender Straße 5
Zweiter Donnerstag im Monat, 18.30 Uhr

Heidelberg

Luzia Dietsche
Tel.: 062 21/54 45 27
luzia.dietsche@urz.uni-heidelberg.de
China-Restaurant »Palast«
Lessingstraße 36
Letzter Mittwoch im Monat, 20.00 Uhr

Karlsruhe

Klaus Braune
Tel.: 07 21/6 08 40 31
braune@rz.uni-karlsruhe.de
Universität Karlsruhe, Rechenzentrum
Zirkel 2, 3. OG, Raum 316
Erster Donnerstag im Monat, 19.30 Uhr

Kiel

Karsten Heymann

karsten.heyman@gmx.de

Letzter Donnerstag im Monat, 19.00 Uhr

Ort siehe <http://zaubrerer.net/latexwiki>

Köln

Helmut Siebert

Institut für Kristallographie

Zülpicher Straße 49b

Letzter Mittwoch im Monat, 19.30 Uhr

München

Michael Niedermair

<http://www.uwe->

siart.de/typografie/stammtisch.html

Gaststätte »Scheidegger«

Bauerstraße 16

Erster Mittwoch im Monat, 19.00 Uhr

Münster

Johannes Reese

reese@linguist.de

Gaststätte »Sabroso«

Mauritzstraße 19

nach Vereinbarung

Stuttgart

Bernd Raichle

bernd.raichle@gmx.de

Bar e Ristorante »Valle«

Geschwister-Scholl-Str. 3

Zweiter Dienstag im Monat, 19.30 Uhr

Trier

Peter Schuster

www.peterfelixschuster.de/tex.htm

Fetzenkneipe (Haus Fetzenreich)

Sichelstraße 36 (beim Sieh-Um-Dich)

54290 Trier

Dritter Montag des Monats, 20.15 Uhr

Ulm

Adelheid Grob

adelan@heidi.in-ulm.de

<http://latex.in-ulm.de>

Gaststätte »Peppers Ulm«

Deinselsgasse 8

Erster Donnerstag im Monat, 19.30 Uhr

Wuppertal

Andreas Schrell

Tel.: 02193/53 10 93

as@schrell.de

Restaurant Croatia »Haus Johannisberg«

Südstraße 10

an der Schwimmooper Wuppertal-Elberfeld

Zweiter Donnerstag im Monat, 19.30 Uhr

Zürich

Johannes Reese

reese@spw.unizh.ch

nach Vereinbarung



Düsseldorf (v.l.): Carsten Binder, Helge Baumann, Georg Verweyen, Christian R.



Freiburg (v.l.): Rolf Niepraschk, Sebastian Inacker, Heiko Oberdiek, Anton Gamel

Adressen

DANTE, Deutschsprachige Anwendervereinigung T_EX e.V.
Postfach 10 18 40
69008 Heidelberg

Tel.: 0 62 21/2 97 66 (Mo, Mi–Fr, 10.00–12.00 Uhr)
Fax: 0 62 21/16 79 06
E-Mail: dante@dante.de

Konten: Volksbank Rhein-Neckar eG
BLZ 670 900 00
Kontonummer 2 310 007
IBAN DE67 6709 0000 0002 3100 07
SWIFT-BIC GENODE61MA2

Postbank Karlsruhe (Auslandsüberweisungen)
BLZ 660 100 75
Kontonummer 213 400 757
IBAN DE93 6601 0075 0213 4007 57
SWIFT-BIC PBNKDEFF

Präsidium

Präsident:	Klaus Höppner	president@dante.de
Vizepräsident:	Volker RW Schaa	vice-president@dante.de
Schatzmeister:	Tobias Sterzl	treasurer@dante.de
Schriftführer:	Manfred Lotz	secretary@dante.de
Beisitzer:	Günter Partosch	
	Bernd Raichle	
	Herbert Voß	

Server

ftp: [ftp.dante.de](ftp://ftp.dante.de)
WWW: <http://www.dante.de/>

Autoren/Organisatoren

Christoph Bier Kahnstraße 5 34233 Fuldatal christoph.bier@web.de	[20]	Michael Niedermair Am Malerwinkel 16 85778 Haimhausen m.g.n@gmx.de	[15]
Wolf-Dieter Busch Talstraße 10 35428 Oberkleen	[32]	Rolf Niepraschk Persiusstr. 12 10245 Berlin Rolf.Niepraschk@ptb.de	[43, 69]
Jürgen Fenn Friedensallee 174/20 63263 Neu-Isenburg juergen.fenn@GMX.DE	[46, 58]	Volker RW Schaa s. Seite 74	[4]
Klaus Höppner s. Seite 74	[4, 5]	Ulrich Schwarz Schweffelstraße 16 24118 Kiel	[8]
Markus Kohm Fichtenstraße 63 68535 Edingen-Neckarhausen Markus.Kohm@gmx.de	[15]	Uwe Siart Aribonenstr. 1 81669 München uwe.siart@ei.tum.de	[57]
Georg Lachenmayr FH Rosenheim Hochschulstraße 1 83024 Rosenheim lachenmayr@fh-rosenheim.de	[5]	Herbert Voß Wasgenstr. 21 14129 Berlin hvoss@tug.org	[3, 57]
Gerd Neugebauer Im Lerchelsbühl 5 64521 Groß-Gerau gene@gerd-neugebauer.de	[33]		

Die T_EXnische Komödie

18. Jahrgang Heft 3/2006 Juli 2006

Impressum

Editorial

Hinter der Bühne

- 4 Grußwort
- 6 Einladung und »Call for Papers« zur Herbsttagung von DANTE e.V.

Bretter, die die Welt bedeuten

- 8 Was hinten rauskommt zählt: Counter Aliasing in L^AT_EX
- 15 Marginalien, da wo man sie haben will!
- 20 typokurz – Einige wichtige typografische Regeln
- 32 DIN-Brief – eine grafische Bedienoberfläche für das L^AT_EX-Paket dinbrief
- 34 Minimal? – Vom Testen eines Textsatzsystems
- 43 Tipps und Tricks: Animierte GIF-Grafiken mit L^AT_EX

T_EX-Beiprogramm

- 46 Neue Pakete auf CTAN

Rezensionen

- 58 Matthias Kalle Dalheimer: *L^AT_EX – kurz & gut*
- 59 »Der L^AT_EX-Begleiter« von Frank Mittelbach, Michel Goossens u. a.

Leserbriefe

- 69 Antwort auf den Leserbrief

Spielplan

- 71 Termine
- 72 Stammtische

Adressen

- 75 Autoren/Organisatoren