

Die T_EXnische Komödie

dante

Deutschsprachige
Anwendervereinigung T_EX e.V.

18. Jahrgang Heft 1/2006 Februar 2006

1/2006

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)

Luzia Dietsche	Gerd Neugebauer	Volker RW Schaa
Hubert Gäßlein	Rolf Niepraschk	Martin Schröder
Rudolf Herrmann	Günter Partosch	
Gert Ingold	Bernd Raichle	

Redaktionsschluss für Heft 2/2006: 15. April 2006

ISSN 1434-5897

Die T_EXnische Komödie 1/2006

Editorial

Liebe Leserinnen und Leser,

was das inhaltliche Erstellen einer Ausgabe von »Die T_EXnische Komödie« angeht, hat Gerd Neugebauer dies schon einmal vor einiger Zeit dargestellt. Normalerweise atmet man als Redakteur erst einmal tief durch, wenn die Datei zur Druckerei geschickt wurde, denn die letzten Tage unterscheiden sich nicht wesentlich von denen anderer Periodika; man läuft in der Regel hinter irgendetwas hinterher, was schon längst da sein sollte, verwechselt ab und zu die Korrekturen und produziert gleichzeitig neue Fehler. Beim letzten Mal musste das Durchatmen verschoben werden. Beabsichtigt war, wie im Editorial auch angegeben, nur eine teilweise farbige Ausgabe. Erfreulicherweise bot die Druckerei zu gleichen Kosten eine komplette Farbausgabe an. Die »fertige« Ausgabe lag da aber schon auf dem FTP-Server der Druckerei. Also noch schnell eine Telefonkonferenz, um eine Abbildung noch nachträglich in Farbe zu bekommen, was an sich unproblematisch erschien, da es eine ganze Seite betraf. Nur konnte der Techniker der Druckerei diese neue Datei nicht öffnen, die bei mir weder unter `xpdf` noch `acroread` Probleme bereitete. »Natürlich« waren die Quellen auf einem Rechner, auf den ich gerade keinen Zugriff hatte und, was eigentlich nicht weiter erwähnt zu werden braucht, das Wochenende nahte ...

Alles kein Problem, könnte man vermuten, doch auch wir haben Auflagen, beispielsweise das pünktliche Mitteilen von Terminen der Mitgliederversammlungen. Kurzum, die entsprechende Seite wurde mehrmals durch den Äther geschickt und war erst von der Druckerei zu lesen, als auf das immer funktionierende PostScript-Format zurückgegriffen wurde. Woran es denn lag, werden vielleicht einige fragen? Das interessiert dann oft keinen mehr, wenn man erst einmal durchgeatmet hat ...

Diese Ausgabe ist zwar wieder formal farblos, inhaltlich aber bunt gemischt, sodass Sie auch diesmal interessante Dinge finden werden. In der Hoffnung, möglichst viele in Berlin begrüßen zu können, verbleibe ich

mit T_EXnischen Grüßen

Ihr Herbert Voß

Die T_EXnische Komödie 1/2006

Hinter der Bühne

Vereinsinternes

Grußwort

Liebe Mitglieder,

diese Ausgabe unserer Mitgliederzeitschrift »Die $\text{T}_{\text{E}}\text{X}$ nische Komödie« halten Sie (hoffentlich) kurz vor der diesjährigen Tagung DANTE 2006 in Händen, die vom 8.–10. März an der Freien Universität Berlin stattfindet. Mittlerweile sind viele Vorschläge für Vorträge und Tutorien eingegangen, dieses Jahr mit einem deutlichen Schwerpunkt auf dem Bereich der Schriften und Anwendungen aus dem $\text{T}_{\text{E}}\text{X}$ -Umfeld (z. B. Grafik und XSL-Verarbeitung). Wir würden uns sehr freuen, wenn Sie sich auch kurzfristig noch zur Teilnahme an der Tagung entschließen können, auch wenn die Hotelsituation in Berlin wegen der zeitgleich stattfindenden Internationalen Tourismusbörse recht angespannt ist.

Während der Tagung findet die übliche Mitgliederversammlung von DANTE e.V. statt, die wie alle zwei Jahre unter dem Hauptaugenmerk der Vorstandswahlen steht. Die gute Nachricht (zumindest für diejenigen, die mit der Arbeit des Vorstands zufrieden sind) ist, dass alle derzeitigen Vorstandsmitglieder eine neue Kandidatur beabsichtigen. Allerdings wird Klaus Höppner als neuer Vorsitzender und Volker RW Schaa als sein Stellvertreter kandidieren. Günter Partosch, der seit 1997 Schriftführer des Vereins ist, möchte seinen Posten aus Zeitgründen einem Nachfolger übergeben und wird als Beisitzer kandidieren. Somit wird es auf der Mitgliederversammlung drei Kandidaten als Beisitzer geben, wobei unsere Satzung bis zu vier Beisitzer zulässt. Für das Amt des Schriftführers hat bereits Manfred Lotz Interesse angemeldet. Weitere Kandidaturen sind natürlich problemlos möglich (insbesondere für diejenigen, die mit der Arbeit des alten Vorstands weniger zufrieden sind).

Ein weiterer Punkt auf der Mitgliederversammlung wird die Option einer kombinierten DANTE-TUG-Mitgliedschaft sein. Dies ist ein Service, den mit der NTG und der UK-TUG bereits zwei europäische $\text{T}_{\text{E}}\text{X}$ -Gruppen anbieten. In diesem Fall könnte ein Mitglied von DANTE zu einem ermäßigten

Preis von 55 \$ (bei derzeitigem Wechselkurs etwa 45 €) zusätzlich Mitglied der weltweiten agierenden T_EX Users Group werden und somit alle regulären Leistungen der TUG (dreimal jährlich das TUGboat) erhalten – mit Ausnahme der jährlichen DVD/CDs, die DANTE-Mitglieder bereits automatisch oder auf Anforderung erhalten. Die rechtliche Eigenständigkeit von DANTE e.V. und TUG bliebe natürlich voll erhalten, während sich das Verfahren für Mitglieder beider Organisation durch Zahlung auf ein deutsches Bankkonto vereinfacht.

Bereits heute sind für alle Mitglieder von DANTE e.V. die Mitgliederpreise für Lizenzen der Lucida-Fonts von Bigelow & Holmes (vielen bekannt aus dem *L^AT_EX companion* bzw. *L^AT_EX-Begleiter*) gültig, die von der TUG unter <http://www.tug.org/store/lucida/> vertrieben werden.

Als nächstes größeres Ereignis steht der Linuxtag an, an dem DANTE e.V. bereits im letzten Jahr mit einem Stand und vielen Vorträgen teilgenommen hat. Die Resonanz war sehr positiv, so dass schnell der Wunsch aufkam, dies dieses Jahr zu wiederholen. Leider gab es wegen Problemen hinter den Kulissen des Linuxtags (Wechsel des Veranstaltungsortes, persönliche Unstimmigkeiten innerhalb des Linuxtag e.V. – anscheinend erwischt es jeden Verein einmal) einige Verzögerungen, mittlerweile ist DANTE e.V. aber in der Liste der Aussteller im Bereich der freien Projekte eingetragen. Endgültiger Termin ist 3.–6. Mai in den Rhein-Main-Hallen in Wiesbaden, somit also früher als üblich (wegen der WM) und an einem anderen Ort. Wir hoffen, dass möglichst viele Freiwillige, ohne die das letztjährige Engagement nicht möglich gewesen wäre, wieder dabei sein werden. Wer sich vom Erfolg des letzten Jahres inspiriert fühlt, ist selbstverständlich herzlich willkommen. Bitte wenden Sie sich in diesem Fall an dante@dante.de, damit wir Sie auf der Mailingliste für den Linuxtag-Stand eintragen können.

Mit freundlichem Gruß,

Volker RW Schaa	Klaus Höppner
Vorsitzender	Stellvertretender Vorsitzender

T_EX-Theatertage

Freude, die ein Maler empfindet

Interview mit Donald E. Knuth¹

Donald Knuth hat mehr als zwei Dutzend Bücher und das Textsatzsystem T_EX geschrieben. Kultstatus unter Informatikern hat ihm jedoch das mehrbändige Lehrbuch »The Art of Computer Programming« eingebracht. Knuth, 1938 in Milwaukee, Wisconsin, geboren, begann mit der Arbeit, noch bevor er 1963 das Studium der Mathematik am California Institute of Technology beendete. Das Werk wurde 1999 von der Wissenschaftszeitschrift »The American Scientist« unter die zwölf bedeutendsten wissenschaftlichen Publikationen des 20. Jahrhunderts eingereiht, ist aber noch immer unvollendet. Technology Review sprach mit Knuth anlässlich der Verleihung der Ehrendoktorwürde an ihn durch die ETH Zürich.

Die Frage wird Ihnen wahrscheinlich sehr oft gestellt. Trotzdem möchten wir natürlich wissen: Wann wird der nächste Band Ihres Buches erscheinen?

Donald Knuth: Ah, ich bekomme die Möglichkeit, Werbung für mein Buch zu machen. Der vierte Band kommt in einer Form heraus, die man Faszikel nennt. Wir Computer-Leute würden dazu eher Beta-Test sagen. Der Plan ist, zweimal im Jahr 125 Seiten als Taschenbuch herauszugeben. Der erste Teil ist im Februar herausgekommen, der zweite im Juli. Nächste Woche schicke ich die 125 Seiten, die im Januar herauskommen sollen, an den Verlag, und ich habe bereits 80 der 125 Seiten geschrieben, die im nächsten Sommer erscheinen sollen. Ich schreibe weiterhin so schnell, wie es mir möglich ist, wobei ich natürlich die von mir gewünschte Qualität beibehalten will.

Natürlich bin ich schon weit über das hinaus, was ich eigentlich als Seitenzahl eingeplant hatte. Aber das, was ich jetzt schreibe, ist so fundamental, dass ich

¹Der Abdruck erfolgt mit freundlicher Genehmigung der Technology Review, des M.I.T. Magazins für Innovation (<http://www.heise.de/tr/aktuell/meldung/66661>). Das Interview fand am 25.11.2005 statt.

einfach die gesamte Geschichte erzählen musste. Wenn ich zu den spezielleren Themen komme, muss ich der Versuchung widerstehen, alles aufzuschreiben.

Das Werk macht also Fortschritte, aber ich muss noch immer lernen, Dinge wegzulassen. Ich sammle nun seit vierzig Jahren Material – mein Haus ist voll von interessanten Notizen –, und ich werde nicht dazu kommen, dies alles zu behandeln, wenn ich mich nicht wirklich ranhalte. Das ist der Grund, weshalb ich nicht viel reise.

Vierzig Jahre sind eine lange Zeit. Hatten Sie je den Eindruck, Sie müssten große Teile des Buches neu schreiben?

Wenn Sie sich die aktuelle Auflage der ersten drei Bände ansehen, dann sind da einige Abschnitte mit einem Symbol markiert, das »Work in Progress« bedeutet. Auf diese Weise habe ich deutlich gemacht, dass ich mir bewusst bin, dass die Dinge sich ändern.

Und wenn Sie beispielsweise den dritten Band nehmen: Da gibt es einen sehr großen Abschnitt rund hundert Seiten über die Datenspeicherung auf Magnetbändern. Die Algorithmen sind sehr interessant, aber die Studenten sind heutzutage nicht mehr motiviert, so etwas zu lernen. Und so wird dieser Abschnitt in der nächsten Ausgabe verschwinden. Wenn ich mit anderen Leuten darüber rede, sagen die mir immer: »Oh, bloß nicht! Ich liebe das.« Aber letztendlich ist das nur ein Drittel von vielen schönen Sachen, die ich rausschmeißen muss.

Sie haben von Schönheit gesprochen und von der Kunst des Programmierens. Können Sie Laien erklären, was ihr Begriff von Schönheit bedeutet?

Das ist wie in der Literatur oder in der Musik. Wenn jemand ein gut geschriebenes Programm liest und den Stil bewundert, dann ist das schön. Sie bewundern vielleicht bestimmte Muster darin oder was auch immer – irgend etwas, das den Teil unseres Gehirns kitzelt, der für Glück zuständig ist. Ich halte heute Abend einen Vortrag über die Freude an technischen Illustrationen und es macht mir einfach Spaß, eineinhalb Stunden dazusitzen und herauszufinden, wie ich eine Abbildung wirklich gut hinbekomme. Wenn ich das schaffe, geht es mir für den Rest des Tages gut.

Das ist dieselbe Art Freude, die ein Maler empfindet oder ein Musiker, der morgens aufwacht und ihm fällt eine Melodie ein, die er dann in sein neues Stück einbaut. Ich kann einen Quelltext lesen und denken, das ist grauenhaft, das passt nicht zusammen, oder aber »Mann, das ist großartig«. Natürlich

kann man über diese Dinge verschiedener Meinung sein, aber das ist auch in der Kunst so.

Es gibt Leute, die sagen, dass die Entwicklung in der Informatik und in der Software-Entwicklung in den vergangenen Jahren viel zu technikzentriert war, dass man die Interaktion zwischen Mensch und Maschine vernachlässigt hat. Was sagen Sie dazu?

Die User wissen nicht immer, was sie wollen. Aber der schlimmste Fehler ist, dass eine Person das System spezifiziert, ein anderer es implementiert und ein dritter es dann benutzen soll, aber diese Leute sich niemals treffen. Das ist besonders schlimm in der ersten Generation eines Systems. Aber eine solche Separation macht die Manager glücklich, denn dann wissen sie, wie sie die Dinge managen sollen. Meiner Auffassung nach sind die erfolgreichsten Software-Projekte in den vergangenen Jahren aber immer mehr anarchisch als hierarchisch strukturiert gewesen. So funktioniert Google, so funktioniert Adobe nun, ich weiß nicht viel über die internen Strukturen von Microsoft. Aber im Allgemeinen stehen die Leute sehr in Verbindung zueinander, und das trotz der Firmen-Rivalität. In der Computing-Szene werden viel mehr Ideen ausgetauscht als beispielsweise in der Biotechnologie. Ich versuche immer, meine Idee von mehr Parties im Silicon Valley zu lancieren, wo Leute aus verschiedenen Firmen hinkommen.

Also propagieren Sie Open Source?

Ich habe mein Textsatzsystem $\text{T}_{\text{E}}\text{X}$ komplett offen gelegt. Aber ich habe immer gesagt, wenn jemand es nehmen und weiterentwickeln will, soll er das gerne tun. Aber er darf es nicht $\text{T}_{\text{E}}\text{X}$ nennen. Ich möchte zu der Idee der Modularität und der Transparenz gerne die Idee der Dauerhaftigkeit stellen feste Punkte zu behalten, etwas, das in zehn Jahren garantiert immer noch dasselbe ist und funktionieren wird.

Hier in Europa gibt es eine andauernde Auseinandersetzung um Softwarepatente. Wie stehen Sie zu dieser Frage? Sind Programme Ihrer Meinung nach patentierbar?

Ich bin ein wenig verwirrt. Als ich im Sommer in Schweden war, hieß es, das Gesetz sei gescheitert. Nun kommt es vielleicht doch.

Es ist jedenfalls gut möglich, dass eine entsprechende Regelung im zweiten Anlauf installiert wird.

Wenn Sie die Anstrengungen gegen die Einführung von Softwarepatenten aufrecht erhalten können, werden meiner Meinung nach eine Menge Leute aus Amerika nach Europa kommen. Wenn es, als ich an T_EX gearbeitet habe, die selben Regelungen gegeben hätte, die es heute in den USA gibt, hätte ich diese Arbeit nie erfolgreich vollenden können.

Ich habe einige Anmerkungen dazu in meinem Buch gemacht – schlagen Sie im Index unter Patente nach. Das größte Problem ist die Trivialität. Da werden Dinge patentiert, die jeder Student in einer Klausur machen kann.

Es gibt Leute, die sagen, dass ein Software-Projekt ab einer bestimmten Größe nicht mehr handhabbar ist. Es gibt andere, die sagen, dieser Punkt sei sowohl für Windows als auch für Linux längst überschritten. Wenn ich Sie richtig verstehe, argumentieren Sie, wenn man sauber dokumentiert und programmiert, kann man noch sehr viel größere Systeme in den Griff bekommen?

Ich bin sicher, dass größere Systeme sicherlich möglich sind. Ich denke aber auch, dass es nicht hoffnungslos wäre, noch einmal ganz von vorne anzufangen. Wenn sich jemand hinsetzen würde und alle bislang bekannten Konzepte von Betriebssystemen noch einmal überdenken würde – ist das beispielsweise die beste Art und Weise Programme zu speichern oder kann man nicht dafür sorgen, dass sich Programme selbst an die jeweiligen Erfordernisse anpassen, wenn sie geladen werden? Ich glaube nicht, dass es wirklich zu spät für so etwas ist. Ich glaube nicht, dass alle guten Ideen der Programmierung bereits entdeckt worden sind.

Ich hoffe natürlich, dass »literate programming« in diesem Projekt irgendwie verwendet wird – irgendeine Form von kombinierter informeller und formeller Beschreibung dessen, was das Programm tun soll, statt es einfach nur zusammenzuhacken, sodass der Computer weiß, was er tun soll.

Eine letzte Frage: Sie haben noch immer keine E-Mail-Adresse und Sie werden sich auch keine mehr zulegen?

Ich habe eine – für die Meldung von Fehlern in meinen Büchern. Das ist sehr wichtig für mich, diese Rückkopplung zu haben. Heute Abend werde ich tatsächlich einen Mann treffen, mit dem ich über Fehlermeldungen korrespondiert habe. Er lebt in der Nähe von Frankfurt und ist nun extra nach Zürich gekommen, um meinen Vortrag zu hören. Wenn der alle Schecks einlösen würde, die er für diese aufgespürten Fehler bekommen hat, wäre er ein reicher Mann. Diese Art der Korrespondenz ist sehr wichtig für mich.

Manche Leute missbrauchen diese Adresse aber auch für andere Zwecke. Meine Sekretärin löscht das Meiste davon. Vor zwei Wochen bekam ich eine E-Mail, in der stand: »Ich bin Student in Korea. Mein Dozent möchte, dass ich recherchiere, was Ihre erste Publikation war. Es ist 'The Art of Computer Programming', richtig?«

Wenn ich auf diese Mail geantwortet hätte, hätte ich zuallererst nach dem Namen seines Dozenten gefragt und dem gesagt, dass er komplett versagt hat.

Aber ein Projekt wie Linux wäre ohne Internet und E-Mail nicht zustande gekommen?

Nun, ich sage nicht, dass alle E-Mail aufgeben sollten. Ich habe mich entschieden, E-Mail aufzugeben. Die Rolle mancher Menschen ist es, an der Spitze der Dinge zu stehen. Meine Rolle ist es, an der Basis zu stehen. Mein Talent, langfristig zu denken, erfordert es einfach, einen ganzen Tag über etwas nachdenken zu können, ohne dass das Telefon klingelt oder ich von sonst etwas abgelenkt werde.

Tatsächlich schicke ich auch E-Mails über meine Sekretärin, wenn ich beispielsweise jemanden um einen Gefallen bitte. Aber wenn ich E-Mail hätte, ich würde morgens zur Arbeit kommen und um 12 wäre ich dann mal soweit, dass ich mir überlegen könnte, was mache ich denn heute.

Bretter, die die Welt bedeuten

Farbig hinterlegte Kopfzeilen mit KOMA-Script

Markus Kohm

Nachdem im Juli 2005 in `scrpage2` die Möglichkeit eingebaut wurde, die horizontalen Linien im Kopf und Fuß der Seite einzufärben, war die logische nächste Frage der Anwender: »Kann man auch den ganzen Kopf mit einer Farbe hinterlegen?« Auch wenn der Autor dieses Artikels der Meinung ist, dass Linien und Farben im Kopf diesem ein zu hohes Gewicht geben, so lautet die Antwort: »Selbstverständlich!«

Schnelle Lösung

Die am häufigsten verwendete Schnittstelle bei `scrpage2` dürfte inzwischen die des vordefinierten Seitenstils `scrheadings` sein. Die Frage ist nun also, wie man mit Hilfe der in Abschnitt 4.1.1, »Vordefinierte Seitenstile«, der KOMA-Script-Anleitung (beispielsweise [3, Seite 146–150]) erklärten Anweisungen eine farbige Hinterlegung des Kopfes erreichen kann.

Zunächst einmal benötigt man dafür ein Paket, das Farbe in die $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Welt bringt. Hier bietet sich die Verwendung des sehr leistungsfähigen `xcolor`-Pakets an (siehe [2]). Als Farbe für den Kopf wird im Beispiel ein helles Lachs verwendet:

```
3 \usepackage[svgnames]{xcolor}
4 \colorlet{headbackground}{LightSalmon}
```

Des Weiteren muss natürlich das `scrpage2`-Paket geladen und initialisiert werden.

```
5 \usepackage{scrpage2}
6 \pagestyle{scrheadings}
7 \clearscrheadfoot
```



Abbildung 1: Nicht durchgängig hinterlegte Kopfzeile

Das vom `xcolor`-Paket verwendete `color`-Paket bietet für das Setzen einer Box mit eingefärbtem Hintergrund die Anweisungen `\colorbox` und `\fcolorbox`. Letztere erlaubt es, zusätzlich eine farbige Umrandung zu setzen. Für den Kopf bietet sich die `\colorbox` an.

Würde man nun beispielsweise innerhalb von `\ihead` einfach ein `\colorbox` verwenden, also

```

9 \ihead{\colorbox{headbackground}{Links}}
10 \chead{\colorbox{headbackground}{Mitte}}
11 \ohead{\colorbox{headbackground}{Rechts}}
```

schreiben, so wäre die Kopfzeile wie in Abbildung 1 nicht durchgehend mit Farbe hinterlegt.

Als Möglichkeit, die Farbe durchgängig zu setzen, bietet sich an, die Box im linken Feld auf die gesamte Breite auszudehnen. Das mittlere und das rechte Feld benötigen dann keine farbliche Hinterlegung mehr, sondern können direkt auf den farbigen Hintergrund des linken Feldes ausgegeben werden. Nun bietet die `\colorbox`-Anweisung keine Option, um die Breite der Box anzugeben. Es ist aber natürlich möglich, in diese Box eine weitere Box mit definierter Breite zu setzen. Hierfür kann beispielsweise eine `\makebox` verwendet werden.

```

15 \ihead{%
16   \colorbox{headbackground}{\hspace{-\fboxsep}}%
17   \makebox[\linewidth][1]{Links}%
18   \hspace{-\fboxsep}}%
19 }%
20 }
21 \chead{Mitte}
22 \ohead{Rechts}
```



Abbildung 2: Durchgängig hinterlegte Kopfzeile

Das erste optionale Argument von `\makebox` gibt die Breite an. Innerhalb des Kopfes gibt `\linewidth` bei `scrpage2` immer die Breite des gesamten Kopfes an. Damit der Text innerhalb der Box linksbündig bleibt, wird als zweites optionales Argument die Ausrichtung 1 gesetzt. Näheres zu den optionalen Argumenten von `\makebox` ist beispielsweise [1] zu entnehmen.

Die beiden `\hspace`-Anweisungen sind wichtig, weil `xcolor` bei `\colorbox` rechts und links immer zusätzlich einen Abstand von `\fboxsep` einfügt, der mit diesen beiden Anweisungen wieder entfernt wird. Das Ergebnis ist Abbildung 2 zu entnehmen.

So richtig gut sieht das aber nicht aus. Das wird jedoch rasch besser, wenn man den Kopf mit

```
26 \setheadwidth{page}
```

auf die gesamte Breite ausdehnt. Es ist dabei sinnvoll, das die Felder trotzdem einen gewissen Abstand zum Rand einhalten.

```
27 \ihead{%
28   \colorbox{headbackground}{\hspace{-\fboxsep}%
29     \makebox[\linewidth][1]{\quad Links}%
30     \hspace{-\fboxsep}%
31   }%
32 }
33 \chead{Mitte}
34 \ohead{Rechts\quad}
```

Abbildung 3 zeigt das verbesserte Ergebnis.



Abbildung 3: Durchgängig hinterlegte Kopfzeile über die gesamte Seitenbreite

Mängel der schnellen Lösung

Nun ergibt sich aber als zusätzliche Problematik, dass im doppelseitigen Druck gar nicht immer das Argument von `\ihead` als erstes gesetzt wird. Auf geraden, also linken, Seiten wird dieses Element zuletzt gesetzt. Das würde bedeuten, dass die Hintergrundfarbe dieses Elements den Inhalt der anderen Elemente überdeckt. Außerdem will man auf linken Seiten den Text im inneren Element dann nicht links- sondern rechtsbündig gesetzt haben.

Das `scrpage2`-Paket bietet zur Lösung dieser Probleme die Anweisungen `\lehead` und `\rehead` sowie `\lohead` und `\rohead`.

Verbesserte Lösung

Nun wäre es natürlich umständlich und unschön, wenn man nur wegen des farbigen Hintergrundes auf `\ihead` und `\ohead` verzichten müsste. Deshalb ist es sicher sinnvoll, die entsprechenden Anweisungen für Kopfzeilen mit farbigem Hintergrund neu zu definieren. Die Schwierigkeit besteht dabei allein darin, dass die entsprechenden `scrpage2`-Anweisungen jeweils ein optionales Argument besitzen, das natürlich bei der Farbversion ebenfalls funktionieren soll. Es muss daher jeweils mit `\@ifnextchar` geprüft werden, ob ein optionales Argument folgt oder nicht, um dann später die `scrpage2`-Anweisungen ebenfalls mit oder ohne optionalem Argument aufzurufen.

```

1 \documentclass[a6paper,twoside]{scrartcl}
2 \usepackage[ngerman]{babel}
3 \usepackage[svgnames]{xcolor}
4 \colorlet{headbackground}{LightSalmon}
5 \usepackage{scrpage2}
6 \pagestyle{scrheadings}
7 \clearscrheadfoot
8

```

```

9 \makeatletter
10 \newcommand{\cbihead}{%
11   \@ifnextchar[{\cb@ihead}{\cb@ihead}}
12 \newcommand{\cb@ihead}[1]{%
13   \rehead{#1}\cblohead{#1}}
14 \newcommand{\cb@ihead}[2][{}]{%
15   \rehead[{}]{#2}\cb@lohead[{}]{#2}}
16
17 \newcommand{\cbohead}{%
18   \@ifnextchar[{\cb@ohead}{\cb@ohead}}
19 \newcommand{\cb@ohead}[1]{%
20   \cblehead{#1}\rohead{#1}}
21 \newcommand{\cb@ohead}[2][{}]{%
22   \cb@lehead[{}]{#2}\rohead[{}]{#2}}
23
24 \newcommand{\cblohead}{%
25   \@ifnextchar[{\cb@lohead}{\cb@lohead}}
26 \newcommand{\cb@lohead}[1]{%
27   \lohead{\colorbox{headbackground}{%
28     \hspace{-\fboxsep}%
29     \makebox[\linewidth][1]{#1}%
30     \hspace{-\fboxsep}}}%
31 }
32 \newcommand{\cb@lohead}[2][{}]{%
33   \lohead[{\colorbox{headbackground}{%
34     \hspace{-\fboxsep}%
35     \makebox[\linewidth][1]{#1}%
36     \hspace{-\fboxsep}}}%
37   {\colorbox{headbackground}{%
38     \hspace{-\fboxsep}%
39     \makebox[\linewidth][1]{#1}%
40     \hspace{-\fboxsep}}}%
41 }
42
43 \newcommand{\cblehead}{%
44   \@ifnextchar[{\cb@lehead}{\cb@lehead}}
45 \newcommand{\cb@lehead}[1]{%
46   \lehead{\colorbox{headbackground}{%
47     \hspace{-\fboxsep}%
48     \makebox[\linewidth][1]{#1}%
49     \hspace{-\fboxsep}}}%
50 }
51 \newcommand{\cb@lehead}[2][{}]{%
52   \lehead[{\colorbox{headbackground}{%
53     \hspace{-\fboxsep}%
54     \makebox[\linewidth][1]{#1}%
55     \hspace{-\fboxsep}}}%
56   {\colorbox{headbackground}{%
57     \hspace{-\fboxsep}%

```

```

58     \makebox[\linewidth][l]{#1}%
59     \hspace{-\fboxsep}}}%
60 }
61 \makeatother
62
63 \begin{document}
64 \cbihead{Innen}
65 \chead{Mitte}
66 \cbohead{Außen}
67 \noindent Testseite Testseite Testseite
68 Testseite Testseite Testseite\clearpage
69 \noindent Testseite Testseite Testseite Testseite
70 Testseite Testseite Testseite Testseite\clearpage
106 \end{document}

```

Die Anweisungen `\cbohead`, `\cbihead`, `\cblehead` und `\cblohead` funktionieren dabei wie die Anweisungen `\ohead`, `\ihead`, `\lehead` und `\lohead` aus `scrpage2`, nur setzen sie zusätzlich den farbigen Hintergrund. Dieser erscheint auch auf Seiten im plain-Stil, falls das optionale Argument einer der Anweisungen verwendet wird. Ein Beispiel für den Kopf einer Doppelseite zeigt Abbildung 4.

Selbstdefinierte Seitenstile

Das Paket `scrpage2` bietet bekanntlich auch die Möglichkeit, eigene Seitenstile zu definieren, statt nur die vordefinierten Seitenstile zu konfigurieren. Die für die vordefinierten Seitenstile gezeigten Möglichkeiten existieren jedoch grundsätzlich ebenso bei Verwendung von `\deftripstyle`, `\defpagestyle`, `\newpagestyle`, `\renewpagestyle` und `\providepagestyle`.

Bei `\deftripstyle` hat man erneut das Problem, dass man im doppelseitigen Druck wieder nur ein Feld für innen und ein Feld für außen hat, wobei auf der linken Seite das äußere Feld und auf der rechten Seite das innere zuerst ausgegeben wird. Man muss hier gegebenenfalls mit

Innen	Mitte	Außen	Außen	Mitte	Innen
Testseite Testseite Testseite Testseite Testseite			Testseite Testseite Testseite Testseite Testseite		

Abbildung 4: Durchgängig hinterlegte Kopfzeile einer Doppelseite


```
1 \ifodd\value{page} ...\else ...\fi
```

testen, ob man sich gerade auf einer ungeraden, also rechten, oder einer geraden, also linken, Seite befindet. Dieser Test funktioniert hier trotz der asynchronen Ausgabe von \TeX , weil bei der Auswertung der Kopf- oder Fußdefinition des Seitenstils die Seitenzahl bereits feststeht.

Bei Verwendung der Anweisungen der Expertenschnittstelle besteht das Problem mit der Wahl des richtigen Feldes für den Hintergrund nicht, weil dort ohnehin unterschiedliche Definitionen für gerade und ungerade Seiten und für Seiten im einseitigen Layout verwendet werden. Eventuell ist deshalb die Verwendung der Expertenschnittstelle die einfachste Lösung. Mit einem kleinen Trick, nämlich durch Angabe einer Breite von 0pt bei `\makebox`, funktioniert das Einfärben hier sogar besonders gut.

```
72 \newpagestyle{coloredhead}{%
73   {\makebox[0pt][l]{%
74     \color{headbackground}%
75     \rule[-\dp\strutbox]{\linewidth}{\headheight}}%
76   \makebox[0pt][l]{Au\ss en}%
77   \makebox[\linewidth]{Mitte}%
78   \makebox[0pt][r]{Innen}%
79   }%
80   {\makebox[0pt][l]{%
81     \color{headbackground}%
82     \rule[-\dp\strutbox]{\linewidth}{\headheight}}%
83   \makebox[0pt][l]{Innen}%
84   \makebox[\linewidth]{Mitte}%
85   \makebox[0pt][r]{Au\ss en}%
86   }%
87   {\makebox[0pt][l]{%
88     \color{headbackground}%
89     \rule[-\dp\strutbox]{\linewidth}{\headheight}}%
90   \makebox[0pt][l]{Links}%
91   \makebox[\linewidth]{Mitte}%
92   \makebox[0pt][r]{Rechts}%
93   }%
94 }{%
95   {\pagemark\hfill}%
96   {\hfill\pagemark}%
97   {\hfill\pagemark\hfill}%
98 }
99 \pagestyle{coloredhead}
```

Prinzipiell kann man diesen Trick auch bei den Lösungen aus den vorherigen Abschnitten verwenden. Allerdings muss man dann selbst bei allen Elementen für die korrekte vertikale Ausrichtung sorgen. Bei der Verwendung der Expertenschnittstelle bekommt man die korrekte vertikale Ausrichtung automatisch.

Schlussbemerkung

Wie in diesem Artikel gezeigt wurde, sind mit `scrpage2` auch farbig hinterlegte Kopfzeilen möglich. Die gezeigten Lösungen sind nicht ganz trivial, aber auch nicht überaus kompliziert. Die angewendeten Prinzipien sind nicht auf die Verwendung von `scrpage2` beschränkt, sondern können auch auf die Verwendung beispielsweise von `fancyhdr` übertragen werden. Doch ganz egal, wie einfach oder kompliziert der Anwender die Lösung nun finden mag: Nicht alles, was machbar ist, sollte automatisch auch gemacht werden.

Literatur

- [1] L^AT_EX3 Project Team: *L^AT_EX 2_ε for authors*; Juli 2001; CTAN:/tex-archive/macros/latex/base/.
- [2] Uwe Kern: *Extending L^AT_EX's color facilities: the xcolor*; Juli 2004; CTAN:/tex-archive/macros/latex/contrib/xcolor.
- [3] Markus Kohm und Jens-Uwe Morawski: *KOMA-Script*; Lehmanns Fachbuchhandlung; Hamburg; 2005; ISBN 3-86541-089-8.

Trennhilfen

Ulrike Fischer

Markieren von Trennstellen und Einfügen von Bindestrichen

In der Dokumentation von `babel.sty` wird zu `"-` erklärt, es sei »an explicit hyphen sign«. Dies ist falsch, `"-` fügt normalerweise keinen Bindestrich ein. Dieser Fehler hat mich animiert, einmal die Trenn- und Bindestrichbefehle genauer unter die Lupe zu nehmen.

Die Befehle mit `"` benötigen `babel.sty` mit der Option `(n)german` oder `(n)german.sty`.

- `\-` Markiert eine mögliche Trennstelle. Wenn dort getrennt wird, wird ein Bindestrich (-) eingefügt. \TeX sucht im Wort *nicht* nach weiteren Trennstellen.
- `-` Fügt einen Bindestrich ein. Ohne weitere Trennhilfen ist er die einzige Trennstelle.
- `--` Fügt einen »Von-Bis-Strich« ein (im Deutschen meist auch als Gedankenstrich verwendet). Ohne weitere Trennhilfen ist er in einem Wort die einzige Trennstelle.
- `---` Fügt einen langen Gedankenstrich ein. Ohne weitere Trennhilfen ist er in einem Wort die einzige Trennstelle.
- `$$` Fügt ein Minuszeichen ein. Keine Trennstelle im Textmodus. Sollte man im Text nicht verwenden.
- `{}-{}$` Fügt ein Minuszeichen als Relationszeichen ein. Danach kann getrennt werden. Sollte man im Text nicht verwenden.
- `"-` Markiert eine mögliche Trennstelle. Wenn dort getrennt wird, wird ein Bindestrich (-) eingefügt. \TeX sucht im Wort noch nach anderen Trennstellen vor und nach dem `"-`.
- `"=` Fügt einen Bindestrich ein. Markiert eine mögliche Trennstelle. \TeX sucht auch nach anderen Trennstellen vor und nach dem `"=`. Es kann an dieser Trennstelle unerwünschte Leerzeilen geben. Nützlich für lange, zusammengesetzte Wörter, beispielsweise `NRW`"=Landtagsfraktion.

"" Markiert eine mögliche Trennstelle. Wenn dort getrennt wird, wird *kein* Bindestrich eingefügt. \TeX sucht im Wort noch nach anderen Trennstellen vor und nach dem "". Es kann an dieser Trennstelle unerwünschte Leerzeilen geben. Bei -"" wird vor dem Bindestrich nicht getrennt!

Nützlich zur Markierung einer Trennstelle nach »/«:
Lehrer/""Lehrerinnen.

`\hspace{Opt}` Alternative zu "".

"" Fügt einen Bindestrich ein, an dem nicht getrennt werden kann. Man kann auch `\mbox{-}` verwenden. \TeX sucht danach *nicht* mehr nach Trennstellen im Wort. Sollen dennoch Worttrennungen erlaubt sein, muss z. B. mit "- eine Trennstelle angegeben werden.

Nützlich z. B. für einen Bindestrich am Anfang eines Wortes:
Weinglas und "-flasche.

Beispiele

Es wird absichtlich eine fehlerhafte Trennstelle markiert, um sicherzustellen, dass sich nicht der automatische Trennalgorithmus einmisch.

Links ist die Eingabe, rechts der normale Ausdruck. Darunter wurden verschieden breite `\parbox` verwendet. Da das erste Wort einer `\parbox` grundsätzlich nicht getrennt wird (außer man fügt davor `\hspace{Opt}` ein), wurde jeweils *Minister* an den Anfang geschrieben.

Ministerkonferenz

Ministerkonferenz

<i>Minister</i>
Mi-
nis-
ter-
kon-
fe-
renz

<i>Minister</i>
Minis-
ter-
konfe-
renz

Ministe\r-konferenz

Ministerkonferenz

<i>Minister</i>
Ministe- rkonferenz

<i>Minister</i>
Ministe- rkonferenz

Ministe-rkonferenz

Ministe-rkonferenz

<i>Minister</i>
Ministe- rkonferenz

<i>Minister</i>
Ministe- rkonferenz

Ministe--rkonferenz

Ministe-rkonferenz

<i>Minister</i>
Ministe- rkonferenz

<i>Minister</i>
Ministe- rkonferenz

Ministe---rkonferenz

Ministe—rkonferenz

<i>Minister</i>
Ministe— rkonferenz

<i>Minister</i>
Ministe— rkonferenz

Ministe\$-rkonferenz

Ministe—rkonferenz

<i>Minister</i>
Ministe-rkonferenz

<i>Minister</i>
Ministe-rkonferenz

Ministe\${}-rkonferenz

Ministe—rkonferenz

<i>Minister</i>
Ministe-rkonferenz

<i>Minister</i>
Ministe-rkonferenz

Ministe\$-{}-rkonferenz

Ministe—rkonferenz

<i>Minister</i>
Ministe-rkonferenz

<i>Minister</i>
Ministe-rkonferenz

Ministe\$-{}\$rkonferenz

Ministe – rkonferenz

<i>Minister</i>
Ministe–
rkonferenz

<i>Minister</i>
Ministe–
rkonferenz

Ministe"-rkonferenz

Ministerkonferenz

<i>Minister</i>
Mi-
nis-
te-
rkon-
fe-
renz

<i>Minister</i>
Mi-
niste-
rkon-
ferenz

Ministe""rkonferenz

Ministerkonferenz

<i>Minister</i>
Mi-
nis-
te
rkon-
fe-
renz

<i>Minister</i>
Mi-
niste
rkon-
ferenz

Ministe\hspace{0pt}rkonferenz

Ministerkonferenz

<i>Minister</i>
Mi-
nis-
te
rkon-
fe-
renz

<i>Minister</i>
Mi-
niste
rkon-
ferenz

Ministe"=rkonferenz

Ministe-rkonferenz

<i>Minister</i>
Mi-
nis-
te-
rkon-
fe-
renz

<i>Minister</i>
Mi-
niste-
rkon-
ferenz

Ministe-""rkonferenz

Ministe-rkonferenz

<i>Minister</i>
Ministe-
rkon-
fe-
renz

<i>Minister</i>
Ministe-
rkon-
ferenz

Ministe/"rkonferenz

Ministe/rkonferenz

<i>Minister</i>
Mi-
nis-
te/
rkon-
fe-
renz

<i>Minister</i>
Mi-
niste/
rkon-
ferenz

Ministe"-rkonferenz

Ministe-rkonferenz

<i>Minister</i>
Ministe-rkonferenz

<i>Minister</i>
Ministe-rkonferenz

Ministe\mbox{-}rkonferenz

Ministe-rkonferenz

<i>Minister</i>
Ministe-rkonferenz

<i>Minister</i>
Ministe-rkonferenz

Ministe"~r"-konferenz

Ministe-rkonferenz

<i>Minister</i>
Ministe-r-
kon-
fe-
renz

<i>Minister</i>
Ministe-r-
konfe-
renz

Mi"-niste"~r"-konferenz

Ministe-rkonferenz

<i>Minister</i>
Mi-
niste-r-
kon-
fe-
renz

<i>Minister</i>
Mi-
niste-r-
konfe-
renz

Mi"-niste"~r"-konferenz

Ministe-rkonferenz

<i>Minister</i>
Mi-
niste-r-
kon-
fe-
renz

<i>Minister</i>
Mi-
niste-r-
konfe-
renz

-konferenz

-konferenz

<i>Minister</i>
-
konferenz

<i>Minister</i>
-
konferenz

-kon"-ferenz

-konferenz

<i>Minister</i>
-
kon-
fe-
renz

<i>Minister</i>
-konfe-
renz

"~konferenz

-konferenz

<i>Minister</i>
-konferenz

<i>Minister</i>
-konferenz

\mbox{-}konferenz

-konferenz

<i>Minister</i>
-konferenz

<i>Minister</i>
-konferenz

"~kon"-ferenz

-konferenz

<i>Minister</i>
-kon-
fe-
renz

<i>Minister</i>
-konfe-
renz

Die GaPFill-Methode zur Erzeugung von L^AT_EX-picture-Umgebungen

Herbert Möller

Geometrie- oder Zeichenprogramme und Perl-Filter werden benutzt, um auch komplizierte Figuren bequem mit der L^AT_EX-picture-Umgebung zu erzeugen. Die Filterprogramme verarbeiten PostScript-Dateien und liefern unmittelbar verwendbaren L^AT_EX-Code. Die Methode wird an zwei Filterprogrammen für das Geometrieprogramm Cabri-géomètre II erläutert. Das erste setzt nur das Paket `ebezier` voraus und ist deshalb unabhängig von einem Treiber. Das zweite unterstützt außerdem das neue Paket `pict2e`.

Einführung

Durch das Makropaket `pict2e` [5] von Hubert Gäßlein und Rolf Niepraschk werden die schwerwiegendsten Beschränkungen der L^AT_EX-picture-Umgebung aufgehoben. Damit bleibt im Wesentlichen das Problem der Objektpositionierung. Ein wegweisender Vorschlag unter Verwendung des

Makropakets P \TeX von Michael J. Wichura [13] wurde 1999 von Robert W.D. Nickalls mit dem MS-DOS-Filterprogramm `mathsPIC` [9] vorgestellt, das er und Apostolos Syropoulos als Perl-Filterprogramm weiterentwickelt haben [10]. Allerdings wird dabei die bequemere Eingabe mit rund 90 zusätzlichen Befehlen erkaufte.

Deshalb stellte sich die Frage, ob die Erzeugung von \LaTeX -Figuren auch ohne Steuerbefehle möglich ist, weil die von einem Grafikprogramm stammenden PostScript-Dateien, die mit dem \LaTeX -Befehl `\special` genutzt werden, offenbar alle benötigten Informationen enthalten. Die Antwort ist eine Methode, die aus den folgenden vier Schritten besteht:

1. Erzeugung der gewünschten Figur mit einem Geometrie- oder Zeichenprogramm;
2. »Drucken« (oder Exportieren) der Konstruktion als PostScript-Datei;
3. Anwenden eines Filterprogramms auf den PostScript-Text;
4. Kopieren der benötigten Teile des Ausgabetextes in ein \LaTeX -Dokument.

Wegen dieser Struktur heißt die Methode »GaPFilL« (*Graphics as PostScript Filtered for \LaTeX*) [8]. Da die Schritte 1. und 2. sowie die im \LaTeX -Dokument verwendeten Pakete variieren, sind verschiedene Filterprogramme nötig. Die Zusammenhänge werden hier am Beispiel von zwei Prototypen beschrieben: `CABebez.pl` setzt nur das Makropaket `ebezier` [2] voraus, `CABpict.pl` benutzt außerdem das oben erwähnte Paket `pict2e`.

Das Geometrie- oder Zeichenprogramm

Es sind alle Programme geeignet, die grafische Objekte erzeugen. Für Diagramme und Illustrationen genügen Zeichenprogramme, die zu Office-Paketen gehören. Das sehr gute Zeichenprogramm `Draw` von `OpenOffice.org` ist sogar kostenlos. Sollen geometrische Sachverhalte dargestellt werden, so ist Geometrie-Software günstiger, weil sie zusätzlich zur Objektherstellung vielfältige Transformationen und Verknüpfungen bietet, wie etwa Spiegelung, Winkelhalbierung, Maßübertragung und Ortslinienherzeugung.

Im Bildungsbereich ist `Cabri-géomètre II` (im Folgenden: `Cabri-Geometrie`) das leistungsfähigste und verbreitetste »dynamische« Geometrieprogramm

für die Betriebssysteme Windows und MacOS. Es wurde seit 1987 von Jean-Marie Laborde und Franck Bellemain am Institut d'Informatique et Mathématiques Appliquées der Universität Joseph Fourier in Grenoble entwickelt. Seine Verbreitung verdankt es unter anderem zahlreichen Fortbildungsveranstaltungen für LehrerInnen sowie der Implementierung einer angepassten Version in den Taschencomputern TI-92, Voyage 200 und weiteren Rechnern von Texas Instruments.

Da also relativ viele Menschen mit diesem Programm vertraut sind, sollen hier nur die Besonderheiten wiedergegeben werden, die bei der Erzeugung von L^AT_EX-Figuren eine Rolle spielen. Seit 2004 gibt es sowohl für Windows-Rechner als auch für Macintosh-Computer die verbesserte Version **Cabri-géomètre II plus**. Auf die Unterschiede wird im Folgenden an den entsprechenden Stellen eingegangen.

Verwendung von Farben

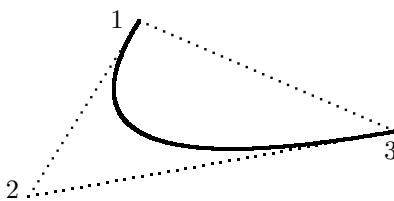
Mit Hilfe von Farben werden einige Zusatzinformationen an das Filterprogramm übermittelt. Die im Folgenden verwendeten Farbnamen stammen aus der deutschsprachigen Macintosh-Version von **Cabri-Geometrie**. Dort stehen nur elf Farben zur Verfügung. In der Mac OS-Version von **Cabri-Geometrie II plus** hat die Farbpalette 36 Farben. In beiden neuen Versionen kann man die RGB-Werte der Farben selber festlegen. Auf Seite 34 wird beschrieben, wie die Farbzuordnung im Filterprogramm zu ändern ist.

Das Färben von Figuren in der L^AT_EX-picture-Umgebung erfolgt bei Bedarf als abschließende Bearbeitung mit Befehlen aus dem **color**-Paket. Das ist aber nur mit Einschränkungen möglich, weil T_EX auf Grund einer technischen Schwierigkeit bei jedem Farbwechsel möglicherweise vertikalen Zwischenraum einfügt (siehe [3], Seite 6). Ohne komplizierte Korrekturen lassen sich deshalb nur Objekte färben, bei denen alle zusammenhängenden Bildteile dieselbe Farbe haben.

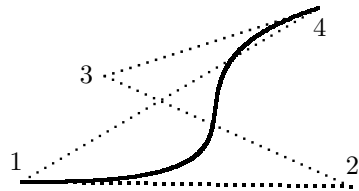
- o Alle *Geraden, Strahlen, Strecken, Kreise* und *Kegelschnitte*, die *gelb* gefärbt sind, werden von dem Filterprogramm ignoriert und dienen deshalb als *Hilfslinien*. Auch die von **Cabri-Geometrie** gezeigten roten Punkte berücksichtigt das Filterprogramm nicht. Geraden, Strahlen, Parabeln und Hyperbeln werden durch **Cabri-Geometrie** am Rand des Bildausschnitts abgeschnitten. Da das Filterprogramm die genaue »bounding box« be-

rechnet, sollten sie nur als Hilfslinien verwendet werden, wenn ihre Form nicht die gewünschte ist.

- *Durchgezogene Strecken* und *Polygone* sind *grün*.
- *Pfeile* werden mit der Farbe *violett* gezeichnet.
- *Gepunktete Strecken*, *Pfeillinien*, *Polygone*, *Kreise* und *Kreisbögen* erhält man mit der Farbe *dunkelgrün*.
- *Quadratische* und *kubische Bézier-Kurven* werden durch *blaue* Polygone mit drei beziehungsweise vier Ecken eingegeben (siehe die Figuren 1 und 2). Die zugehörigen Kurvenstücke lassen sich bei *Cabri-Geometrie* mit Hilfe von Makros sichtbar machen. Die Konstruktion solcher Makros ist in Abschnitt 4 der Dokumentation [8] beschrieben.



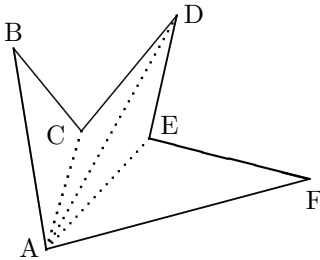
Figur 1



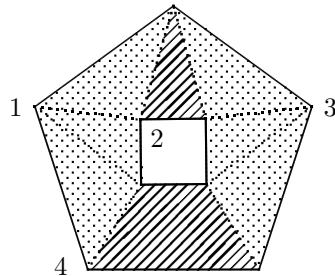
Figur 2

- *Polygone* sind durch Anklicken des Ausgangspunktes oder durch Doppelklicken zu *schließen*. Durch Verwenden der Farbe *marineblau* wird erreicht, dass die letzte Strecke nicht erscheint, so dass sich auch (nicht geschlossene) *Streckenzüge* bequem eingeben lassen.
- Zur *Positionierung von Text* beziehungsweise Formeln werden Zweipunktpolygone mit der Farbe *blau* verwendet. Damit sich die »bounding box« korrekt berechnen lässt, sollte die markierende Strecke ungefähr die links unten beginnende Diagonale des kleinsten Rechtecks sein, das den Text beziehungsweise die Formel umschließt. An der Position des Anfangspunkts der Strecke erscheint in der \LaTeX -Ausgabe eine fortlaufende Nummer.
- Die übrigen fünf Farben werden zum *Füllen*, *Schraffieren* und *Punktieren* von Flächen benutzt, die von *Polygonen* berandet sind. Das Filterprogramm zerlegt dazu die von einem Polygon gebildete Fläche in Dreiecke mit dem Ausgangspunkt des Polygons als gemeinsamem Eckpunkt. Um Überschneidungen zu vermeiden, ist also die gegebene Fläche durch Polygone so zu unterteilen, dass für jedes Polygon beim Verbinden des

Ausgangspunkts mit den übrigen Eckpunkten jeweils Dreiecke entstehen, die höchstens eine Seite mit jedem der anderen Dreiecke gemeinsam haben (siehe die Figuren 3 und 4).



Figur 3



Figur 4

Rote Polygone werden ohne Rand gefüllt. Zum Schraffieren mit sichtbaren Randstreifen ist die Polygonfarbe *orange* zu wählen, ohne Randstreifen die Farbe *braun*. Hat das Polygon die Farbe *purpur*, so wird die Fläche punktiert und die Randstreifen sind sichtbar; mit der Farbe *violett* lässt sich die Fläche ohne Rand punktieren. Die bisherigen Farbuordnungen sind in Tabelle 1 zusammengefasst.

- Da *Viertel-* und *Halbkreise*, die in L^AT_EX auch mit dem `\oval`-Befehl zusammenhängen, in *Cabri-Geometrie* als Kreisbögen mit Hilfe von drei Punkten auf einem Kreis zu konstruieren sind, ermöglichen gefärbte *Kreise* die bequemere Eingabe dieser Kreisbögen mit achsenparallelen Endpunktradien. Tabelle 2 enthält die zugeordneten Farben. Bei der Macintosh-Version von *Cabri-Geometrie* lässt sich die jeweilige Objektfarbe im »Standardfenster« voreinstellen.

PostScript-Dateien

Die extreme Geräteabhängigkeit der Einbindung von Grafiken mit dem L^AT_EX-Befehl `\special` wird durch die Berücksichtigung der wichtigsten Ausgabetreiber zur Erzeugung von PostScript-Code bei der Definition geeigneter T_EX-Makros erheblich gemildert. Die L^AT_EX-Pakete `graphicx`, `color` und `pstricks` sind Beispiele hierfür. Sie werden neben anderen Möglichkeiten in [6] beschrieben. Die Vielfalt der PStricks-Makros lässt sich z. B. aus [12] entnehmen.

Tabelle 1: Allgemeine Farbuordnungen

Typ	Farbe
Hilfslinie	gelb
durchgezogenes Objekt	grün
Pfeil	violett
gepunktetes Objekt	dunkelgrün
Bezier-Kurve oder Text	blau
Streckenzug	marineblau
ohne Rand füllen	rot
mit Rand schraffieren	orange
ohne Rand schraffieren	braun
mit Rand punktieren	purpur
ohne Rand punktieren	dunkelbraun

Tabelle 2: Farbuordnungen bei Kreisteilen

Typ	Farbe
Hilfskreis	gelb
durchgezogener Kreis	marineblau
punktierter Bogen oder Kreis	dunkelgrün
linker Halbkreis	purpur
rechter Halbkreis	rot
unterer Halbkreis	orange
oberer Halbkreis	dunkelbraun
linker unterer Viertelkreis	blau
linker oberer Viertelkreis	grün
rechter unterer Viertelkreis	braun
rechter oberer Viertelkreis	violett

Bei der Programmierung von Filterprogrammen zur Extraktion von L^AT_EX-relevanten Daten aus EPS-Dateien (EPS = Encapsulated PostScript) braucht man wesentlich geringere Kenntnisse – etwa aus [1] – als bei der Herstellung von entsprechenden T_EX-Makros. Für den Einsatz mit einem passenden Filterprogramm kann man einen Standard-EPS-Druckertreiber verwenden, ohne etwas von PostScript zu verstehen. Geeignete Treiber lassen sich kostenlos von der Treiberseite der Firma Adobe (<http://www.adobe.com/support/>

downloads/main.html) herunterladen. In [4] wird die Einrichtung eines EPS-Druckertreibers unter Windows 95 beschrieben.

Die folgenden Erläuterungen sind für diejenigen gedacht, die ein Filterprogramm ändern oder neu schreiben wollen. PostScript-Dateien können mit Editor-Programmen wie *Alpha* oder *BEdit* unter MacOS oder *WinEdt* unter Windows geöffnet und bearbeitet werden. Von den häufig mehr als 1000 Zeilen enthält nur ein sehr kleiner Teil, der zwischen `EndProlog` und `EOF` liegt, \LaTeX -relevante Daten.

Für die Anwendung von Filterprogrammen ist es günstig, dass alle Zeilen mit benötigten Zahlenwerten eine eindeutige Kennzeichnung am Zeilenende enthalten, die allerdings bei den einzelnen PostScript-Erzeugern recht verschieden sein kann. Die Zeilen mit den Farbdaten enden unter anderem mit `setrgbcolor` oder mit `:F4`. Die Objektdaten folgen ein bis drei Zeilen später. Auch hier kann die Kennzeichnung »konkret« oder »abstrakt« sein. Im Falle einer Strecke ist sie z. B. `lineto stroke` beziehungsweise `@b`. Im zweiten Falle lässt sich die konkrete Zuordnung im Prolog-Teil finden.

Die Bedeutung und die Reihenfolge der Zahlenwerte sind bei verschiedenen PostScript-Erzeugern ebenfalls nicht einheitlich. Zum Beispiel kann ein Kreis durch zwei Eckpunkte seiner »bounding box« oder durch seinen Mittelpunkt und Radius bestimmt sein. Manchmal sind die x - und y -Koordinaten von Punkten vertauscht. Schließlich gehören auch die Zahlenwerte nicht immer zu derselben Einheit. Meistens ist sie `pt` und passt damit gut zur `picture`-Umgebung von \LaTeX . Es können auch ganzzahlige Werte auftreten, die mit 0,25 zu multiplizieren sind, um die Point-Größe zu erhalten.

Alle diese Unterschiede lassen sich durch systematisches Probieren klären. Dazu wird jeweils ein Objekt mit dem Zeichen- oder Geometrieprogramm hergestellt und nach dem Notieren der Maße in eine PostScript-Datei gedruckt. Die meisten Grafikprogramme geben Koordinaten und Längen in Millimetern an. Der Umrechnungsfaktor zu `pt` ist 2,845. Durch die Wahl von geeigneten Gitterpunkten oder durch Variation der Figur kann dann die Position der gesuchten Daten bestimmt werden. Stellt es sich heraus, dass die y -Koordinaten in der PostScript-Datei entgegengesetzt zu denen im Grafikprogramm sind, so muss entweder jede Grafik vor dem Abspeichern vertikal gespiegelt werden, oder im Filterprogramm ist jede benötigte y -Koordinate mit `-1` zu multiplizieren.

Perl

Die Interpretersprache Perl (als Abkürzung von »**P**ractical **E**xtraction and **R**eport **L**anguage«) wurde um 1988 von Larry Wall zunächst zum Scannen von beliebigen Textdateien entwickelt, um damit Informationen zu extrahieren und zu verarbeiten. Sie kann als jüngerer Bruder des um 1978 von Donald E. Knuth veröffentlichten \TeX -Systems angesehen werden, weil seither viele Personen zur Vervollkommnung beitragen und weil jedes der beiden Systeme mit zahlreichen Modulen und Hilfen im »Comprehensive \TeX Archive Network« (CTAN) beziehungsweise im »Comprehensive Perl Archive Network« (<http://www.perl.com/CPAN>) kostenlos zur Verfügung gestellt wird. Inzwischen ist Perl eine universelle Script-Sprache, mit der viele Arten von wiederkehrenden Aufgaben automatisiert werden können, z. B. im Systemmanagement von Computern und ihrer Peripheriegeräte. Neben der großen Leistungsfähigkeit seit Version 5 ist die Verfügbarkeit auf allen verbreiteten Plattformen besonders vorteilhaft für den Einsatz im Zusammenhang mit \LaTeX . Es ist deshalb nicht überraschend, dass im CTAN schon zahlreiche Perl-Programme zu finden sind.

Wer eine höhere Programmiersprache (wie z. B. C) kennt, wird kaum Schwierigkeiten haben, an Hand von Beispielen und mit Hilfe einiger der rund 70 Online-Handbucheile Perl-Filterprogramme, wie die in dem folgenden Abschnitt beschriebenen, anzufertigen oder zu ändern. Ähnlich wie bei \LaTeX ist zu erwarten, dass sich nach der derzeit erreichten Perl-Versionsnummer 5.8.7 in dem für uns in Frage kommenden Bereich des Textfilterns nichts Wesentliches mehr ändern wird. Deshalb ist es sinnvoll, jetzt auch für weitere Kombinationen von Grafik- oder Geometrieprogrammen, PostScript-Versionen und \LaTeX -Paketen Perl-Filterprogramme bereitzustellen und zu pflegen. Das kann im Interesse der \LaTeX -Gemeinde von wenigen Personen durchgeführt werden, weil die Hauptarbeit mit der Entwicklung der Filterstruktur und der Unterprozeduren geleistet ist.

Die Filterprogramme CABpict.pl und CABebez.pl

Das 32 kB große CABpict.pl [8] ist 964 Zeilen lang – einschließlich der Kommentare, die mit `\#` beginnen und der Zeilen mit schließenden Klammern `\}` – und CABebez.pl hat nur 29 Zeilen weniger. Deshalb werden im Folgenden höchstens Zeilen und Blöcke kommentiert, die für das Verständnis wichtig sind oder die bei Änderungen eine Rolle spielen können.

Bei Übereinstimmung werden die Zeilen aus CABpict.pl genommen. Die Zeilennummern, die in beiden Fällen nicht zum Programm gehören, können dabei verschieden sein. In diesem Bericht sind die Zeilen von CABebez.pl durch einen Doppelpunkt hinter der Nummer gekennzeichnet.

```

1  #!/perl -w
2  # CABpict.pl
3  # (c) Copyright 2006 H. M"oller (mollerh@math.uni-muenster.de).
4  # Version 1.1 for Cabri-g\`eom\`etre II with MacOS 9.x, ...
5  # This program may be distributed and/or modified under the conditions of the
   LaTeX Project Public License,
6  [ ... ]
7  use POSIX( 'ceil', 'floor' );

```

Die erste Zeile, die wegen des ! unmittelbar nach dem #-Zeichen, keinen Kommentar darstellt, fehlt in einigen Perl-Programmen, z. B. wenn der Aufruf durch eine Kommandozeileneingabe erfolgt oder wenn keine Optionen verwendet werden. Die Option `-w` veranlasst Perl alle Fehlermeldungen auszugeben. Es kann auch notwendig sein, dass vor `perl` ein Pfadname steht, z. B. `/usr/bin/`.

Die Zeilen 2 bis 6 sind für Versions- und Lizenzangaben vorgesehen. Durch Zeile 7 werden zwei Prozeduren aus dem Perl-Modul `POSIX` bereitgestellt. Sie dienen zum Runden und lassen sich (mit Vorsicht) durch eigene (Unter-)Prozeduren ersetzen, die den Aufruf `int` verwenden.

```

12 # Unitlength in pt:
13 $sul = 1.0;
14 # Fill factor (for filling with magnification up to 500%)
15 $fillf = 5;
16 # Point factor:
17 $pointf = 0.3;
18 # Flag for dotting parabolic arcs (1: Dotting)
19 $Qbezflag = 0;
20
21 # Bezier factor:
22 $bezf = 2.0;

```

Jeder dieser fünf Parameter kann vor der Anwendung des jeweiligen Programms geändert werden. Sie sollten danach wieder ihre ursprünglichen Werte bekommen, es sei denn, das Programm wurde unter einem anderen Namen gespeichert.

Durch Variation der `unitlength $sul` lässt sich jede Figur ausgehend von der PostScript-Darstellung in der L^AT_EX-picture-Umgebung verkleinern oder

vergrößern. Der Wert $\$ul = 1.0$ ergibt die Figur in der Originalgröße aus Cabri-Geometrie.

Der Füllfaktor $\$fillf$ regelt den Abstand der waagerechten Strecken beim Füllen von Polygonen. Mit $\$fillf = 5$ erscheinen Figuren bei einer Bildschirmvergrößerung bis zu 500% vollständig gefüllt.

Mit `pict2e` allein lassen sich keine punktierten Strecken und Kurven darstellen, weil `pict2e` die Punkteanzahl von Bézier-Objekten ignoriert. Deshalb werden in `CABpict.pl` die Plotbefehle `\Lbezier` und `\Qbezier` aus dem Paket `ebezier` hinzugenommen. Dabei ergibt der Punktfaktor $\$pointf = 0.3$ den normalen Punktabstand (wie in den Figuren 1 bis 3). Um eine Kurve zu punktieren, kann man bei gesetztem Schalter $\$Qbezflag = 1$ mit Parabelbögen approximieren, für die eventuell in der \LaTeX -picture-Umgebung bei `\Qbezier` die Punkteanzahl anzupassen ist. Im Normalfall $\$Qbezflag = 0$ wird der Parabelbogen mit dem Plotbefehl `\qbezier` von `pict2e` gezeichnet.

In `CABbez.pl` ergibt der \>Bézier factor\< $\$bezf = 2.0$ einen Punktabstand, der bei normaler Vergrößerung und im Druck die entsprechende Linie geschlossen erscheinen lässt. Da der Speicherbedarf von Bézier-Kurven bei `ebezier` nicht selten achtmal so groß ist wie bei `pict2e`, kann $\$bezf$ bei knappem Speicher verkleinert werden.

```
22 # Color names:
23 $yellow = "0.9843900.9511410.020249";
24 [ ... ]
```

Cabri-Geometrie gibt die drei RGB-Werte in der PostScript-Datei siebenstellig mit einer Ziffer vor dem Dezimalpunkt aus. Durch die Zeilen 45 bis 47 sucht `CABpict.pl` diese mit `_setrgbcolor_` endenden Zeilen und entfernt die Leerzeichen sowie das Schlusswort `setrgbcolor`. Damit ergibt die jeweilige Zeile ein einziges »Wort«, das später den entsprechenden Zeichenketten vorangestellt wird. Durch die Zuordnung von einprägsamen Farbnamen zu diesen »Ziffernworten« ist es einfach, andere oder weitere Farbnamen zu verwenden.

```
38 # Constants in dotted figures:
39 $uli = sp(4 / $ul);
40 $ule = sp(0.8 / $ul);
```

```
38 # Constant in cubic Bézier curves for quarters of a circle:
39 $190 = "0.552284749830794";
```

Um die Flächen punktieren zu können, die in beliebigen geschlossenen Polygonen enthalten sind, haben die Punkte in Bezug auf die jeweilige `picture`-Umgebung absolute Koordinaten (x, y) , wobei x und y gerade ganze Zahlen sind und $x+y$ durch 4 teilbar ist. `$uli` liefert dann den waagerechten Abstand und `$ule` die Länge der 0.8pt dicken Strecken, die einen Punkt bilden.

Die Zahlkonstante `$190` wird bei der Approximation von Kreisbögen durch kubische Bézier-Kurven benötigt. Ihre Herleitung ist in [2] (Seite 6) zu finden.

```

42 @lines = <>;
43 do {
44   $_ = $lines[$i++];
45   if (/ setrgbcolor\s/o) {
46     s/ //go;
47     s/setrgbcolor\s/ /o;
48     $c = $_;
49     $_ = $lines[$i++];
50     s/ moveto/o;
51     s/lineto stroke/stroke/o;
52     s/curveto stroke/curveto/o;
53     s/ setlinewidth stroke//o;
54     s/ lineto//go;
55     if (/stroke/o) {
56       $line[++$#line] = $c.$_;
57     }
58     elsif (/closepath fill/o) {
59       $vector[++$#vector] = $c.$_;
60     }
61     elsif (/arc /o) {
62       $circle[++$#circle] = $c.$_;
63     }
64     elsif (/arcn/o) {
65       $arc[++$#arc] = $c.$_;
66     }
67     elsif (/curveto/o) {
68       do {
69         $conic[++$#conic] = $c.$_;
70         $_ = $lines[$i++];
71         s/ moveto/o;
72         s/curveto stroke/curveto/o;
73       }
74       until $_ !~ /curveto/o;
75     }
76   }
77 }
78 until $i == $#lines;

```

Mit den Zeilen 42 bis 78 werden die benötigten Daten aus der PostScript-Datei extrahiert, wobei das Array `@lines` zunächst alle Zeilen enthält. Die sukzessive Zuweisung jeder einzelnen Zeile zu der allgemeinen »last-result-Variablen« `$_` erfolgt in Zeile 44. Wie oben beschrieben, werden dann die »Farbzeilen« gesucht und komprimiert. Zeile 48 weist das Ergebnis der Variablen `$c` zu.

In der vorliegenden Version braucht anschließend nur die nächste Zeile analysiert zu werden. Die Suchen-und-Ersetzen-Befehle der Zeilen 50 bis 54 sorgen dabei für Eindeutigkeit und für reine Zahlensequenzen (bis auf das Schlusswort). Bei Versionen, in denen die Daten erst in einer späteren Zeile enthalten sind, ist es zweckmäßig, zunächst die Zwischenzeilen und die Datenzeile durch Entfernen des Zeilensprungs zu einer Zeile zusammenzufügen.

In Abhängigkeit von den verschiedenen Zeilenendungen werden dann die Daten in fünf Arrays gesammelt, wobei z. B. Zeile 56 bedeutet, dass das Array `@line`, dessen letztes Element den Index `$#line` hat, um ein Feld verlängert wird, das die Zeichenkette der Koordinaten des Anfangs- und des Endpunkts einer Strecke mit vorangestellter Farbvariablen `$c` aufnimmt.

Unabhängig von der Färbung eines Pfeils schreibt **Cabri-Geometrie** die Koordinaten der vier Eckpunkte einer zu füllenden PSTricks-Pfeilspitze in die PostScript-Datei. Mit `CABpict.pl` werden diese Daten nicht benötigt, wenn die Pfeilfarbe violett ist, weil dann der Plotbefehl `\vector` zur Verfügung steht. Andernfalls wird die Pfeilspitze aus zwei gefüllten Dreiecken zusammengesetzt und die Pfeillinie je nach Farbe als durchgezogene oder gepunktete Strecke gezeichnet.

Die Kreisdaten sind in den mit `arc_` endenden Zeilen zu finden, wobei zusätzlich 0 bzw. 360 als vierter und fünfter Zahlwert einen geschlossenen Kreis charakterisieren. Dadurch lassen sich auch Kreisbögen erkennen, die in **Cabri-Geometrie** im Uhrzeigersinn eingegeben wurden, weil sie in Zeilen, die mit `arc_` enden, an Stelle von 0 und 360 den Anfangs- und den Endwinkel enthalten. Die Zeilen mit den Daten von 'Links'-Kreisbögen schließen mit `arcn`.

In der Macintosh-Version von **Cabri-Geometrie** besteht die Möglichkeit, durch fünf Punkte festgelegte Kegelschnitte zu zeichnen. In der zugehörigen PostScript-Datei werden dazu die Daten von approximierenden kubischen Bézier-Kurven gespeichert und zwar neun bei Ellipsen und vier bei jedem offenen Zweig. Die Werte aus den entsprechenden Zeilen, die am Ende

durch `curveto stroke` gekennzeichnet sind, werden in dem Array `@conic` gespeichert. Diese Daten können unmittelbar mit dem Plotbefehl `\cbezier` verarbeitet werden.

```

80 $pflag = 1;
81 $sflag = 1;
82 $thicknessflag = 1;
83 $coun = 0;
84 $xtex = "";
85 $mtex = "";
86 $bmax = 500;
87 [ ... ]
88 $btex = "\\documentclass{article}\n\\usepackage{e bezier}\n\n";

```

Diese Initialisierungen beginnen mit Schaltern für die Startecke von Polygonen, für die Startwerte der »bounding box« und für die jeweils eingestellte Strichdicke. Der Parameter `$coun` ergibt die Nummern für die Text- und Formelstellen. Mit `$xtex` wird die Zeichenkette aufgebaut, die schließlich durch einen `print`-Befehl den L^AT_EX-Text im Perl-Ausgabefenster liefert. Da die Textmarkierungen in jedem Fall nachzubearbeiten sind, wird durch Zwischenspeicherung in `$mtex` dafür gesorgt, dass alle diese Textstellen leicht auffindbar am Ende der Ausgabe erscheinen.

Werden mit dem Paket `e bezier` Bézier-Kurven mit mehr als 500 Punkten erzeugt, so ist es notwendig, den L^AT_EX-Wert 500 von `\qbeziermax` zu vergrößern. Deshalb wird mit der Variablen `$bmax` die maximale Punkteanzahl bestimmt und beim Überschreiten von 500 in der Zeichenkette `$btex`, die den Anfang des entsprechenden L^AT_EX-Programms enthält, der Wert von `\qbeziermax` mit `\renewcommand` angepasst. Die Einbindung von `$btex` und `$mtex` in `$xtex` findet sich auf Seite 40.

```

87 # Lines and polygons
88 $cflag = 1;
89 foreach (@line) {
90     @coo = split;
91     $co0 = $coo[0];
92     $co2 = (-1) * $coo[2];
93     $co4 = (-1) * $coo[4];
94     if (($co0 ne $violet) and ($co0 ne $yellow)) {
95         if ($cflag) {
96             $xtex .= "%Lines, arrows, polygons and Bézier curves\n";
97             $cflag = 0;
98         }
99         if ($co0 ne $blue) {
100             bound($coo[1], $co2);

```

```

101     bound($coo[3],$co4);
102   }
103   if (($co0 ne $red) and ($co0 ne $blue) and ($co0 ne $brown)
104       and ($co0 ne $darkbrown) and ($co0 ne $navy)) {
105     lin($co0,$coo[1],$co2,$coo[3],$co4);

```

Hier beginnt die Verarbeitung der Daten aus den Arrays. Da diese längeren Teile bei den meisten Kombinationen von Grafikprogrammen, PostScript-Versionen und \LaTeX -Paketen eine ähnliche Struktur haben, werden im Folgenden nur wesentliche oder typische Ausschnitte kommentiert.

Der Schalter `$cflag` sorgt dafür, dass sich im \LaTeX -Programm die Kommentare, die auf die jeweils entstehenden Objekte hinweisen, nicht ständig wiederholen. Mit dem `split`-Befehl werden die durch `foreach` gelieferten Zeichenketten in Listen von Zeichenketten umgewandelt, die in unserem Falle die Zahlen und die Schlusswörter der ausgewerteten PostScript-Zeilen sind. In der hier benutzten Kurzform von `split` bewirken Leerzeichen die Trennung der Zeichenkette des last-result-Werts `$_`.

Da die meisten Verzweigungen von mindestens einer Farbe abhängen, wird zunächst das Listenelement mit dem Index 0, das die Farbe enthält, abgekürzt. Anschließend erfolgt die in dieser Version notwendige »vertikale Spiegelung« durch Multiplikation aller y -Koordinaten mit -1 .

Zeile 96 ergibt eine typische \LaTeX -Kommentarzeile, weil die zu `$xtex` durch `.=` hinzugefügte Zeichenkette mit dem `%`-Zeichen beginnt und mit dem Zeilensprungbefehl `\n` von Perl endet.

Mit den Zeilen 100 und 101 wird jeweils die Unterprozedur zur Bestimmung der »bounding-box«-Koordinaten aufgerufen. Sie ist in den Zeilen 944 bis 957 definiert. In diesem Fall werden die Koordinaten des Anfangs- und Endpunkts von Strecken ausgewertet. Objekte mit der Farbe `blue` sind ausgenommen, weil die Diagonale bei der Text- und Formelmarkierung ein Zweipunktpolygon ist und weil bei Bézier-Kurven 51 Punkte berücksichtigt werden.

`bound` stellt eine von neun Unterprozeduren dar, die ab Zeile 390 (356 bei `CABebez.pl`) zu finden ist, wobei die Reihenfolge keine Rolle spielt. Die meisten dieser Prozeduren werden deshalb bei ihrem ersten Auftreten beschrieben.

Die Unterprozedur `lin` erzeugt alle auszugebenden Strecken. In `CABpict.pl` steht für durchgezogene Strecken der erweiterte `\line`-Befehl von `pict2e` zur Verfügung, während gepunktete Strecken mit dem Befehl `\lbezier` des `ebezier`-Pakets zu zeichnen sind. Bei `CABebez.pl` müssen auch alle Strecken,

die nicht die Steigungs- oder Längenbedingungen von L^AT_EX erfüllen, mit Hilfe von `\l bezier` aufgebaut werden.

```

107   if (($co0 ne $green) and ($co0 ne $darkgreen)) {
108     if ($pflag) {
109       $cb1 = $coo[1];
110       $cb2 = $co2;
111       $pol = $co0." ".$cb1." ".$cb2;
112       $pflag = 0;
113     }
114     else {
115       $pol .= " ".$coo[1]." ".$co2;
116       if (abs($coo[3] - $cb1) + abs($co4 - $cb2) < 2.0E-6) {
117         $poly[++$#poly] = $pol;
118         $pflag = 1;

```

Für jedes Polygon, das gefüllt, gepunktet oder schraffiert werden soll, entsteht hier das Array `@poly`, das die Koordinaten aller Ecken enthält.

```

1   @p = best(abs($dy / $dx));
2   $psx = sp($p[1]) * ($dx <=> 0);
3   $psy = sp($p[0]) * ($dy <=> 0);

```

Für die Ausgabe von durchgezogenen Strecken beziehungsweise von vollständigen Pfeilen mit dem Paket `pict2e` wird durch `best` eine Unterprozedur bereitgestellt, die Zähler und Nenner der rationalen Steigungsnaherungen bestmöglich mit Hilfe eines Kettenbruchalgorithmus liefert. Da die beiden Komponenten insbesondere teilerfremd sind, stellt es praktisch keine Einschränkung dar, dass sie in dem Intervall $[-1000, 1000]$ liegen müssen.

Mit der zweizeiligen Unterprozedur `sp` werden die Zahlen, die Cabri-Geometrie mit sechs Stellen hinter dem Komma an die PostScript-Datei übergibt, sowie alle übrigen in der L^AT_EX-picture-Umgebung erscheinenden Dezimalzahlen auf drei Stellen nach dem Dezimalpunkt mit Beseitigung von Schlussnullen gerundet.

```

150  foreach (@poly) {
151    @po = split;
152    $p0 = $po[0];
153    $pon = $#po;
154    if (($p0 eq $red) or ($p0 eq $purple) or ($p0 eq $darkbrown)
155        or ($p0 eq $orange) or ($p0 eq $brown)) {

```

Bei der Auswertung der Daten des Arrays `@poly` spielt neben der Farbe der Index `$pon = $#po` des letzten Elements eine Rolle. Ist `$pon = 6`, so wird die

Unterprozedur `tri` aufgerufen, die das Füllen, Punktieren und Schraffieren von Dreiecken erledigt. Im Falle `$pon = 8` lassen sich achsenparallele Rechtecke mit Hilfe der Unterprozedur `rect` einfacher behandeln als alle übrigen Polygone, die zunächst in Dreiecke zerlegt werden. Bei der Farbe `$blue` ergibt `$pon = 4` die Textmarkierungen, während `$pon = 6` und `$pon = 8` zum Aufruf der Unterprozeduren `qbez` für quadratische beziehungsweise `cbez` für kubische Bézier-Kurven führen.

```

214 # Arrows
215 foreach (@vector) {
216   @ve = split;
217   if ($ve[0] eq $darkgreen) {
218
219     tri($red,$vu0,$vu1,$vu2,$vu3,$ve[5],$ve[6]);
220     tri($red,$vu0,$vu1,$vu6,$vu7,$ve[5],$ve[6]);

```

Wie oben schon erwähnt, wird hier bei Pfeilen mit einer gepunkteten Pfeilinie (und stets bei Pfeilen in `CABebez.pl`) die Pfeilspitze aus zwei gefüllten Dreiecken zusammengesetzt.

Kreisbögen werden in Viertelkreise und in einen kürzeren Restbogen zerlegt. Bei `CABebez.pl` approximiert die Unterprozedur `quart` alle Viertelkreise (und damit auch Kreise und Halbkreise) durch eine kubische Bézier-Kurve. Mit `pict2e` stehen die von Einschränkungen befreiten Plotbefehle `\oval` und `\circle` zur Verfügung. Für alle übrigen Kreisbögen liefert die Unterprozedur `arc` die Approximation durch eine kubische Bézier-Kurve. Um in `CABpict.pl` gepunktete Kreise zu erhalten, wird der Plotbefehl `\cCircle` des Pakets `ebezier` verwendet.

```

379 # Frame
380 if ($xtex . $mtex ne "") {
381   $xtex = "\\documentclass{article}\n\\usepackage{ebezier}\n".
382     "\\usepackage[pdfTeX,pstArrows]{pict2e}\n\n \\begin{document}\n\n".
383     "\\setlength{\\unitlength}{". $ul. "pt}\n".
384     "\\begin{picture}(" . ceil(($xmax - $xmin)) . "," .
385     ceil(($ymax - $ymin)) . ")(" . floor($xmin) . "," . floor($ymin) . ") \n".
386     "\\linethickness{0.8pt}\n". "\\thicklines\n". $xtex;
387   $xtex .= $mtex. "\\end{picture}\n\n\\end{document}";
388 }
389 print $xtex. "\n";

```

```

342 if ($bmax > 500) {
343   $bmax = 100 * ceil($bmax / 100);
344   $btex .= "\\renewcommand{\\qbeziermax}{". $bmax. "}\n";

```



```

345 }
346 if ($xtex . $mtex ne "") {
347 $xtex = $btex. "\\begin{document}\n\n\\setlength{\\unitlength} {\".
348 $ul.\"pt}\n\". "\\begin{picture}(<\".ceil(($xmax - $xmin)).\", \".
349 ceil(($ymax - $ymin)).\")(<\".floor($xmin).\", \". floor($ymin).\")\n\".

```

Hier werden die verschiedenen Teile der Zeichenkette `$xtex` vereinigt und durch den `print`-Befehl in das Perl-Ausgabefenster geschrieben. Da die anschließend zusammengestellten Unterprozeduren weitgehend ungeändert in anderen Versionen verwendet werden können, folgen jetzt nur noch Kommentare, die die zu Grunde liegende Theorie erläutern. Im Hauptteil von `best` wird die eingegebene Dezimalzahl in einen Kettenbruch umgewandelt und dann unter allen Haupt- und Nebennäherungsbrüchen, die »beste Näherungen« sind, derjenige mit dem größtmöglichen Zähler beziehungsweise Nenner bestimmt (siehe [11], Seite 46 und Seite 51).

```

536 # Triangles
537 sub tri {
538 my ($q0, $qx1, $qy1, $qx2, $qy2, $qx3, $qy3) = @_ ;

545     if ($q0 eq $red) {
546 # Filled triangle
547     %ha = ($qx1, $qy1, $qx2+1e-07, $qy2+1e-07, $qx3+2e-07, $qy3+2e-07);
548     @hb = ();
549     @hc = ();
550     foreach (sort { $ha{$a} <=> $ha{$b} } keys %ha) {
551         $hb[++$#hb] = $_;
552         $hc[++$#hc] = $ha[$_];
553     }
554     ($qx1, $qx2, $qx3) = @hb;
555     ($qy1, $qy2, $qy3) = @hc;

```

Jeder der drei Teile von `tri` beginnt mit dem Sortieren der Ecken: Zum Füllen und Punktieren werden sie nach der Größe der y -Koordinate und zum Schraffieren nach der Größe der Differenz beider Koordinaten angeordnet. Bei dem verwendeten Sortierverfahren von Perl für den Datentyp »Hash« ist zu beachten, dass in unserem Fall jedes der beiden assoziierten Arrays aus verschiedenen Elementen bestehen muss. Das wird hier durch Änderung der siebenten Stelle hinter dem Dezimalpunkt erreicht.

Denkt man sich nach dem Sortieren eine Gerade durch den ersten und dritten Eckpunkt gelegt, so lässt sich mit linearer Algebra die Halbebene bestimmen, in der der zweite Punkt liegt. Ausgehend von der Verbindungsstrecke des

ersten und dritten Punkts und mit Fallunterscheidung bezüglich des zweiten Punkts werden dann die einzufügenden Objekte konstruiert. Die Berechnung der Zwischenpunkte für die »bounding box« bei quadratischen und kubischen Bézier-Kurven erfolgt mit den Formeln von Abschnitt 4 der Dokumentation [8]. Die Herleitung der Koeffizienten von kubischen Bézier-Kurven, die in der Unterprozedur `arc` Teilbögen von Viertelkreisen approximieren, ist in [2] (Seite 11 f.) zu finden.

Die `picture`-Umgebung in \LaTeX

Mit der Nutzung der Leistungsfähigkeit von Geometrie- und Zeichenprogrammen werden auch mehrere Vorteile der \LaTeX -`picture`-Umgebung gegenüber dem `\special`-Befehl wirksam:

- Alle Grafikdaten sind in die \TeX -Datei integriert. Insbesondere können bei der Verbreitung keine Grafikdateien verloren gehen.
- Die Flexibilität von \LaTeX zeigt sich vor allem bei der Positionierung durch `minipage`-Einsatz oder in Form von Gleitobjekten. Da die Filterprogramme sowohl die Größe der »bounding box« als auch die Offset-Werte (für die linke obere Ecke) liefern, kann jede Figur direkt in eine `minipage` gelegt und mit dieser z. B. durch Anwendung des `\hspace`- und des `\raisebox`-Befehls horizontal und vertikal verschoben werden (siehe die Figuren 1 bis 4).
- Die Nachbearbeitung von Figuren kann im üblichen Zyklus `edit-typeset-preview` erfolgen.
- Der Speicherbedarf ist selbst bei Dokumenten mit zahlreichen komplizierten Figuren gut kalkulierbar und insbesondere bei der Verwendung von `pict2e` überraschend gering. Aber auch mit dem `ebezier`-Paket sind bei `BIG \TeX` -Versionen keine Speicherprobleme zu erwarten.
- Bei der Erzeugung von PDF-Dateien aus \LaTeX -Vorlagen werden die Figuren ohne weiteres übernommen.

Der Anstoß für die Entwicklung der Filterprogramme kam von dem Wunsch des Autors, rund 80 mit dem Zeichenprogramm `STAD` auf Atari-Computern hergestellte Figuren in ein mit \LaTeX anzufertigendes Buch zu übernehmen. Sie wurden mit dem Zeichenprogramm `Claridraw` »durchgepaust«, als PostScript-Dateien gespeichert und mit einem AppleScript-Programm in \LaTeX -

picture-Umgebungen umgewandelt. Dieses erste funktionierende PostScript-Filterprogramm findet sich in [7] in der Abteilung ›Programme‹.

Literatur

- [1] Adobe, Systems Incorporated: *PostScript Language Reference Manual*; Addison-Wesley Publishing Company; Reading, Mass.; 2. Aufl.; 1995.
- [2] Gerhard A. Bachmaier: *Using ebezier*; 2002; Paket in CTAN:/macros/latex/contrib/ebezier.
- [3] David P. Carlisle: *Packages in the graphics bundle*; 2004; CTAN:/macros/latex/required/graphics/grfguide.pdf.
- [4] Marco Dübendorfer: *Erstellen von EPS-Grafiken unter Windows für die Einbindung in \LaTeX* ; 1999; CTAN:/info/german/grafik/eps_anle.htm.
- [5] Hubert Gäßlein und Rolf Niepraschk: *The pict2e-package*; 2003; Paket in CTAN:/macros/latex/contrib/pict2e/pict2e.dtx.
- [6] Michel Goossens, Sebastian Rahtz und Frank Mittelbach: *The \LaTeX Graphics Companion*; Addison-Wesley Publishing Company; Reading, Mass.; 1997.
- [7] Herbert Möller: *Mathkompass*; Internet: <http://wwwmath1.uni-muenster.de/u/mollerh>.
- [8] Herbert Möller: *Die GaPFilL-Methode zur Erzeugung von \LaTeX -picture-Umgebungen (How to generate \LaTeX picture environments using the GaPFilL method)*; 2006; CTAN:/graphics/gapfill/GaPDoc1.pdf.
- [9] Richard W.D. Nickalls: *MathsPIC*; 1999; CTAN:/graphics/pictex/mathspic.
- [10] Richard W.D. Nickalls und Apostolos Syropoulos: *mathsPIC_{Perl}*; 2005; CTAN:/graphics/pictex/mathspic/perl.
- [11] Oskar Perron: *Die Lehre von den Kettenbrüchen*; Verlag B. G. Teubner; Stuttgart; 3. Aufl.; 1954.
- [12] Herbert Voß: *PSTricks, Grafik mit PostScript für $T_{\text{E}}X$ und \LaTeX* ; Lehmanns/DANTE; 2. Aufl.; 2005.
- [13] Michael J. Wichura: *The P_lCT_EX Manual*; 1992; Paket in CTAN:/graphics/pictex.

T_EX-Beiprogramm

Neue Pakete auf CTAN

Jürgen Fenn

m3D von *Anthony Phan* ist »ein weiteres« Paket, mit dem man dreidimensionale Grafiken mit METAPOST erstellen kann.

CTAN:/graphics/metapost/contrib/macros/m3D

indic-type1 bietet Type1-Fonts, die mit METAPOST erzeugt worden waren, sowie die PFB und MAP-Files für 122 Fonts zu zehn indischen Sprachen (Devanagari, Sanskrit, Gurmukhi/Punjabi, Bangla, Tamil, Sinhala, Malayalam, Kannada, Telugu und Tibetan). Kopien der TFM-Files liegen ebenfalls bei, um die Schriften ohne die Originalpakete zu testen. Weitere Informationen findet man unter <http://www-hep.fzu.cz/~piska/indic.html>. Autor ist *Karel Píška*.

CTAN:/fonts/ps-type1/indic

labelcas ist ein Paket, mit dem man testen kann, ob labels für Querverweise in einem Dokument existieren. Autor ist *Ulrich Diez*.

CTAN:/macros/latex/contrib/labelcas

TengwarScript von *Ignacio Fernández Galván* ist ein Paket, mit dem man auf *Tengwar*-Fonts zugreifen kann.

CTAN:/macros/latex/contrib/tengwarscript

fouriernc enthält L^AT_EX-Mathematikfonts, die die Schriftart *NC Schoolbook* für *Utopia* (erfordert die Fourier-GUTenberg-Fonts).

CTAN:/fonts/fouriernc

arXivBib von *John Forkosh* ist ein Tool, das kurze Zusammenfassungen (*abstracts*) von [arXiv.org](http://arxiv.org) herunterlädt und als $\text{BIB}\text{T}_{\text{E}}\text{X}$ -Einträge formatiert. Näheres ist unter <http://www.forkosh.com/arxivbib.html> zu erfahren.

CTAN:/support/arxivbib

sphyphb sind neue experimentelle Trennmuster für Spanisch, die *Rodrigo Medina* erstellt hat.

CTAN:/language/spanish/nonstandard/sphyphb

hep und *hepthesis* dienen zum Setzen von Arbeiten im Fach Hochenergiephysik. Der Autor ist *Andy Buckley*.

CTAN:/macros/latex/contrib/hep

CTAN:/macros/latex/contrib/hepthesis

stubs ist ein praktisches Makro von *Norbert Stuhmann*, mit dem man die von Aushängen her bekannten Abreißleisten (für Telefonnummern und Ähnliches) in den Fuß einer Seite setzen kann.

CTAN:/macros/latex/contrib/stubs

MetaUML ist eine METAPOST -Bibliothek zum Setzen von UML-Diagrammen (Klassendiagramme, Fallunterscheidungsdiagramme, Statusdiagramme, Ablaufdiagramme). Der Autor ist *Ovidiu Gheorghies*. Die Homepage des Projekts findet man unter: <http://metauml.sourceforge.net> .

CTAN:/graphics/metapost/contrib/macros/metauml

fontch dient zur Änderung der Schriftgröße bei Verwendung von *Plain T_EX*. Unterstützt werden 8, 10, 12, 14, 20 und 24pt. Optional kann auch die Schriftart *Latin Modern* in T1 und in T2 angesprochen werden. Im Mathematikmodus werden aber immer die *Computer Modern*-Schriften verwendet. Der Autor ist *Rodrigo Medina*.

CTAN:/macros/plain/contrib/fontch

fonttable von *Peter Wilson* ist ein Paket, mit dem man alle oder einen Teil der Zeichen in einem beliebigen Dokument als Tabelle ausgeben lassen kann, die in einer Schriftart verfügbar sind.

CTAN:/macros/latex/contrib/fonttable

auncial-new und *blencoding* sind neue Pakete von *Peter Wilson*, der damit die Umstellung der von ihm betreuten alten Schriftarten auf neue Kodierungen fortsetzt. *auncial-new* enthält eine *PostScript-Type1*-Version der Schriftart *Artificial Uncial (auncial)*, einer Handschrift, die während des Hochmittelalters in Gebrauch war. Die Schriftart liegt auch in der neuen, experimentellen Kodierung *B1* vor.

CTAN:/fonts/auncial-new

CTAN:/macros/latex/contrib/blencoding

nddiss ist eine Vorlage für Diplomarbeiten und Dissertationen an der Universität *Notre Dame*, die *Sameer Vijay* bereitgestellt hat.

CTAN:/macros/latex/contrib/nddiss

flacards ist eine neue Klasse von *Norbert Stuhmann*, um »*flashcards*« zu setzen: Mehrere benutzerdefinierte Karten pro Seite werden ausgegeben. Man kann unterschiedliche Inhalte auf der Vorder- und der Rückseite der jeweiligen Karte unterbringen, z. B. um Vokabeln zu üben.

CTAN:/macros/latex/contrib/flacards

har2nat ist ein Paket zur Verwaltung der Bibliographie. Es dient zur Vereinfachung einer Migration von dem Paket *harvard* von *Peter Williams* und *Thorsten Schnier* zu *natbib* von *Patrick W. Daly*, indem es die Zitier-Befehle des Pakets aus *harvard* umdefiniert. Der Autor ist *Bas Straathof*.

CTAN:/macros/latex/contrib/har2nat

minionpro von *Achim Blumensath* dient zur Verwendung der Schriftart *MinionPro* von *Adobe*. Auch griechische, kyrillische, Symbol- und Mathematik-Schriften werden unterstützt.

CTAN:/fonts/MinionPro

pst-calendar ist ein neues Paket von *Manuel Luque* und *Herbert Voß*, das auf *PSTricks* beruht und dazu dient, Kalender für den Zeitraum 2000–2099 zu berechnen und zu plotten.

CTAN:/graphics/pstricks/contrib/pst-calendar

pst-pdf von *Rolf Niepraschk* und *Hubert Gäßlein* vereinfacht die Verwendung von Graphiken, die mit *PSTricks* erstellt worden waren, und anderem PostScript-Code in PDF-Dokumenten. Man greift auf externe Programme zurück, ganz ähnlich wie beim Erstellen einer Bibliographie mit *BIBTEX*.

Das PostScript-Material, einschließlich der Grafiken, wird während des Kompilierens in einer PDF-Datei namens `\PDFcontainer` gesammelt, um diese später anstelle des PS-Codes in ein PDF einzufügen.

CTAN:/graphics/pstricks/contrib/pst-pdf

lshort-vietnamese ist die vietnamesische Übersetzung von *l2kurz*, die Reinhard Kotucha erstellt hat.

CTAN:/info/lshort/vietnamese

Berichtigungen zu »T_EX und die Druckvorstufe«

Martin Schröder

Zwei kleine Korrekturen zum hervorragenden Artikel von Siep Kroonenberg in der DTK 4/2005:

- Der Zähler `\pdfoptionpdfminorversion` funktioniert auch mit neuesten Versionen von pdfT_EX; allerdings wird seit Version 1.30 bei seiner Benutzung darauf hingewiesen, dass es inzwischen den kürzeren Zähler `\pdfminorversion` gibt, der das Gleiche bewirkt.
- Die Beschreibung von `\pdffilemoddate` ist falsch: Der Befehl fügt nicht das Änderungsdatum (»ModDate«) in das Info-Verzeichnis des erzeugten PDF ein, sondern liefert das Änderungsdatum einer Datei im PDF-Datumsformat: Beispielsweise wurde die Eingabedatei für diesen Leserbrief am `\pdffilemoddate{druckvorstufe.ltx}` D:20060129214337+01'00' zuletzt geändert. Die in Kürze erscheinende Version 1.40 von pdfT_EX wird das ModDate automatisch setzen.

Rezensionen

»Schnell ans Ziel mit L^AT_EX 2_ε« von Jörg Knappen (2. Auflage)

Christoph Bier und David Kastrup

»Schnell ans Ziel mit L^AT_EX 2_ε« ist im Sommer 2004 in der 2. völlig überarbeiteten Auflage erschienen. Laut Buchrücken richtet sich das Buch an »alle L^AT_EX-Nutzer, die lange Texte gestalten und Formeln setzen müssen«. Es will einen »kompakten Leitfaden für das wissenschaftliche Arbeiten mit L^AT_EX« darstellen.

Einleitung

Es ist mittlerweile schon einige Monate her, als ich das Buch zwecks Rezension das erste Mal gelesen habe. Mangels Zeit bin ich erstmal nicht über das Notizzettel-Stadium hinausgekommen. Doch das gibt mir nun, bevor ich das Buch noch einmal lese, die Möglichkeit zu fragen: »Was ist bei mir hängen geblieben?«

Ich muss direkt an das Layout denken, das eher unglücklich gewählt ist. Ich erinnere mich an aus meiner Sicht exotische Kapitel (phonetische Symbole, Frakturschriften), und daran dass KOMA-Script nur als »eine interessante Alternative zu den Standardklassen« erwähnt, aber nicht besprochen wird und daran, dass sich kein klares Bild der Zielgruppe einstellte. Leider wird auch BibT_EX nur kurz angesprochen und als nicht lohnenswert »für den einmaligen Gebrauch, etwa beim Schreiben einer Diplom- oder Doktorarbeit« (Seite 23) bezeichnet.

Gestaltung

Direkt beim allerersten Durchblättern fand ich auf Seite 184 einen gravierenden Darstellungsfehler in einer Tabelle, der mich gar dazu veranlasste beim Verlag nachzufragen, ob ich die Version erhalten hatte, die auch in den Buchhandlungen steht. So etwas kann natürlich passieren, schreckt den potenziellen Käufer in der Buchhandlung aber möglicherweise ab.

Ich frage mich außerdem, warum man Daumenregister verwendet, wenn sie nicht einheitlich auf den Seiten erscheinen. Der Abstand zum Rand variiert von Seite zu Seite, was auf mich einen unprofessionellen Eindruck macht und unruhig wirkt. Überhaupt wirkt das gesamte Layout, insbesondere die Verwendung der EC-Schriften, wenig zeitgemäß, eher unschön altmodisch und nicht »rund«. Gliederungsebenen werden in grau dargestellt, wodurch sie naturgemäß etwas verblässen.

Tabellen- und Abbildungsbeschriftungen stehen jeweils im äußeren Rand, in dem ihnen nur für 20–25 Zeichen in der Breite Platz zur Verfügung steht. Dennoch sind sie in Blocksatz gesetzt, was sie häufig löchrig erscheinen lässt und einzelne Beschriftungen reichen auch in den Textbereich hinein. Die Bezeichnungen in Kapitel 2 sind nicht einheitlich, manchmal werden Beispiele als »Bsp.« und manchmal als »Abb.« bezeichnet. Da das Buch ansonsten auch Randbemerkungen verwendet, geht die Übersichtlichkeit, die sie ja eigentlich erhöhen sollen, auf manchen Seiten verloren.

Ob der interessierte Leser in der Buchhandlung das Buch auf den ersten Blick ähnlich wahrnimmt und sich dadurch gar vom Kauf abhalten lässt, vermag ich natürlich nicht zu sagen. Doch für Literaturempfehlungen sind ohnehin der Inhalt des Buches und die Leserführung, also der didaktische Aufbau, das Entscheidende.

Und somit stellt sich die Frage, ob das Buch und für welche Zielgruppe es empfehlenswert ist.

Aufbau und Inhalt

Es geht los mit einem Intro. Darin wird zunächst auf gut vier Seiten kurz die Entstehung von T_EX und L^AT_EX, das Konzept und die Vor- und Nachteile von L^AT_EX beschrieben. Beim Konzept wird der falsche Eindruck erweckt, »Wortprozessoren«¹ unterstützten keine logische Auszeichnung; dies trifft so

¹Gemeint sind Textverarbeitungen à la Word.

allerdings nicht unbedingt zu. Sehr gelungen finde ich die kurze Beschreibung der traditionellen Buchproduktion auf Seite 2, die dem Leser vor Augen führt, dass er mit den üblichen Textverarbeitungen nicht nur Autor, sondern eben auch Gestalter und Setzer ist; Dinge von denen er in der Regel kaum Ahnung hat.

Auf Seite 5 kommt dann auch gleich schon ein erstes Textbeispiel mit L^AT_EX, womit ein Anfänger aber vollkommen überfordert sein dürfte. Er soll das Beispiel in seinem »bevorzugten Texteditor« erstellen »und dann mit dem Befehl `latex`« übersetzen. Die Bedienung des wichtigsten Werkzeugs – des Editors – wird aber an keiner Stelle beschrieben. Der Anfänger, der es gewohnt ist, Word oder OpenOffice.org Writer und Ähnliches zu öffnen und loszutippen, steht hilflos vor der ersten großen Hürde. Schon auf der nächsten Seite wird der Leser ohne weitere Erläuterung mit dem Begriff *Paket* konfrontiert. Als Nächstes wird ihm auf Seite 6 gezeigt, dass Umlaute nicht direkt, sondern über spezielle Befehle eingefügt werden ("‘Hallo sch"one gro"se Welt"’). Spätestens jetzt dürfte mancher Anfänger das Buch frustriert wieder ins Regal stellen – und somit von dem drei Seiten später erwähnten Paket `inputenc` nichts mehr erfahren. Im gleichen Beispiel wird auch noch das laut TLC2 [1] obsoleete Paket `a4` verwendet, das nicht mehr verwendet werden sollte [4].

In **Kapitel 2** geht es dann um den Aufbau eines Dokuments. In einem Unterabschnitt auf Seite 19 wird explizit vor Flattersatz gewarnt, weil »guter Flattersatz [...] schwierig zu erreichen« sei, es wird aber nicht das dafür entwickelte Paket `ragged2e` erwähnt. Zwei Seiten weiter wird dem Leser mitgeteilt, dass er Fußnoten »nur im normalen Fließtext« verwenden kann, aber beispielsweise nicht in Tabellen. Die Lösung für dieses Problem wird dem Leser jedoch vorenthalten.

Die Manipulation des Seitenstils wird auf einer halben Seite besprochen und endet mit einem Verweis auf die Dokumentation von `fancyhdr` (Seite 26). Zur Gestaltung von Folien werden auf der nächsten Seite auch nur wenige Sätze geschrieben und die Klasse `slides` empfohlen. Andere Lösungen, die auch Präsentationen erlauben, finden keine Erwähnung.

Sehr gut finde ich **Kapitel 3**, das die »Feinheiten des Textsatzes« für deutsche und englische Texte beschreibt. Der Leser erfährt, wo Leerzeichen hingehören und wo nicht, und wie er geschützte Leerzeichen, die einen Umbruch an der Stelle verhindern, erhält. Der richtige Satz von Abkürzungen und die richtige Verwendung der verschiedenen Striche und Anführungszeichen wird dargestellt.

Kapitel 4 beschäftigt sich mit Tabellen. Schon nach einer Seite wird der Leser mit den Zählern (`topnumber`) und Anteilen (`topfraction`) zur Steuerung der Platzierung von Gleitumgebungen konfrontiert. Auch dies stellt meines Erachtens zumindest für den Anfänger eine Überforderung dar. Diese Dinge zu erwähnen halte ich zwar für wichtig, aber an anderer Stelle und dann auch etwas ausführlicher. Der Leser wird auch nicht darüber informiert, dass er seine Tabellen nicht zwangsweise in eine Gleitumgebung packen muss. So finden dann auch die Pakete `capt-of` und `caption` keine Erwähnung. Auf die Pakete `dcolumn` und `numprint` wird auch nicht verwiesen. Stattdessen wird eine recht eigenwillige Methode zur Ausrichtung von Zahlen am Dezimaltrenner angeboten, die mit logischer Auszeichnung nicht mehr viel zu tun hat. Ein weiteres Paket, das im Zusammenhang mit Tabellen nicht erwähnt wird, ist `array`. Gut finde ich allerdings die »Hinweise zur Gestaltung von Tabellen« auf Seite 44, die mit `tabsatz` [3] vergleichbar sind und die kurze Beschreibung von `booktabs`.

Das **5. Kapitel** beschreibt »Grafische Effekte im Text«. In diesem Kapitel ist auf einer Seite auch die Beschreibung von `\parbox` und der Umgebung `minipage` untergebracht, und auf mehreren Seiten wird die Verwendung von Farbe mit dem Paket `color` beschrieben; `xcolor` findet keine Erwähnung. Auf vier farbigen Seiten gibt es einige Beispiele für den Einsatz von Farbe, wobei verschiedene Farbkombinationen gut erläutert werden.

Weiter geht es in **Kapitel 6** mit der Verwendung von Bildern. Gleich im ersten Abschnitt wird ein weit verbreitetes Missverständnis provoziert, über das Anfänger immer wieder stolpern und dessen Grundstein bereits in Kapitel 4 gelegt wurde: Weder Tabellen noch Bilder *müssen* in einer der Gleitumgebungen `table` oder `figure` stehen. Gut ist allerdings schon direkt der nächste Abschnitt, in dem der Leser erfährt, wie er ein Abbildungsverzeichnis erhält und wie er auf Abbildungen verweisen kann. Leider ist dies auch gleichzeitig die einzige Stelle im Buch zu Querverweisen (Seite 57).

Als einfachste Art ein Bild einzubinden, wird das Einkleben beschrieben (Seite 58), was einige Verwunderung in mir ausgelöst hat. Dies ist heutzutage sicherlich keineswegs mehr zeitgemäß und ob es die einfachste Art ist, sei auch mal dahingestellt. Der Autor läuft damit meiner Meinung nach Gefahr, L^AT_EX einen altmodischen, überholten Anstrich zu verpassen. Die nächste Möglichkeit, die beschrieben wird, ist das eigene Erstellen von Bildern in der etwas umständlichen Standardumgebung `picture`. `pstricks` und `xypic` werden nur namentlich erwähnt, `pgf` gar nicht. Die Beschreibung der `picture-`

Umgebung fällt zwar etwas knapp aus, doch wer mit den Beispielen etwas herumexperimentiert, versteht sicherlich, was wie funktioniert.

Die Möglichkeiten von `graphicx` zur Manipulation von Farbe, Skalierung und Orientierung normaler L^AT_EX-Konstrukte werden ausreichend behandelt; die insgesamt vier farbigen Beispielseiten schließen auch eine komplette Farbtabelle der mit Namen versehenen `dvips`-Farben ein.

Der Befehl `\includegraphics` wird etwas stiefmütterlich behandelt, denn es wird ausschließlich die Möglichkeit der Einbindung von EPS- und PDF-Dateien erwähnt, kein Beispiel für pdf_TE_X gebracht, und auch nicht darauf hingewiesen, dass man die Dateiendung der eingefügten Datei weglassen kann (und sollte). Auch `\graphicspath`, `\DeclareGraphicsRule` und andere wichtige Details finden keine Erwähnung.

Kapitel 7 zu den Schriften fällt recht ausführlich und aufschlussreich aus. Man findet auch eine Erklärung, warum das Buch in den EC-Schriften gesetzt ist, denn Knappen ist bekanntlich auch Autor dieser Schriften und verbindet damit »Klarheit, Nüchternheit, Lesbarkeit und klassische Eleganz« (Seite 74).

Leider werden in diesem Kapitel auf Seite 75 die veralteten Pakete `palatino`, `times` und `mathptm` erwähnt, die mittlerweile durch `mathpazo` und `mathptmx` ersetzt wurden. `times` bindet im Gegensatz zu `mathptm` und `mathptmx` nicht die passenden Mathematikschriften ein, wobei `mathptmx` der Nachfolger von `mathptm`² ist. Das Gleiche gilt für `palatino` und `mathpazo`. [4]

Der Formelteil ist für seinen Umfang von etwa 40 Seiten gut gemacht. Ein Minuspunkt ist, dass `$` für im laufenden Text gesetzte Formeln gar nicht erwähnt wird. Der Autor beschränkt sich hier auf `\(. . \)`, was aber den Nachteil hat, nicht robust zu sein und deswegen etwa in Überschriften Probleme bereitet. Auch wird der Leser, der Quelltexte von Dritten überarbeitet, häufig `$` als Mathematikbegrenzer finden.

Inkonsequent ist, dass abgesetzte Formeln `\[. . \]` keine Erwähnung finden, sondern hier ausschließlich auf die `displaymath`-Umgebung verwiesen wird, ohne zu erwähnen, dass diese unter `amsmath` nicht verwendet werden soll.

Das sind aber auch schon die wesentlichen Kritikpunkte: Die vorgestellten Mathematikstrukture sind recht vollständig und in sinnvoller Anordnung präsentiert, die Pakete von $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX werden an passender Stelle erwähnt, die unpraktische und mit $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX inkompatible `eqnarray`-Umgebung wird

²Dieses Paket wird in einem späteren Kapitel allerdings erwähnt.

nur dem Namen nach aufgeführt, stattdessen werden die grundlegenden unter den sinnvolleren \mathcal{AMS} - \LaTeX -Konstrukten beschrieben.

Es gibt zwar Kuriositäten wie »Die Buchstaben A–Z, die großen griechischen Buchstaben und die Ziffern 0–9 sind variabel unter \LaTeX .« – was der Autor damit meint, bleibt rätselhaft. Davon abgesehen ist aber sicherlich der Formelteil für einen einführenden Text einer der besseren auf dem Markt: Eine gute Auswahl der präsentierten Pakete und Funktionen, und eine prägnante Darstellung.

Eine besondere Kuriosität bietet der Index im Anhang C, »Schlüssel zum Auffinden von mathematischen Symbolen«: Dieser ist nach Anzahl von »Zügen«, durchgehenden Linien im Symbol, geordnet. Hierbei gibt es dann Unterordnungen wie »Zwei Züge, davon eine gerade Linie mit Ecken«.

In **Kapitel 11** wird die »Planung eines \LaTeX -Projektes« beschrieben. Dem Leser will bei der Auswahl der geeigneten Dokumentklasse geholfen werden, ohne aber zu erwähnen, worin sich die verschiedenen Klassen unterscheiden. Im selben Abschnitt wird auf Seite 121 `\linespread` zur Anpassung des Zeilenabstands genannt, um die Anforderungen einer örtlichen Prüfungsordnung einhalten zu können. Aber leider werden nicht die in diesem Fall (üblicherweise »1,5-facher« Zeilenabstand) meist unerwünschten Nebenwirkungen auf Fußnoten und Listen erwähnt. Für solche Fälle bietet sich das Paket `setspace` an, das diese Nebenwirkungen umgeht. [4]

In Abschnitt 11.3 erfährt der Leser, was er beachten muss, wenn er seinen \LaTeX -Quelltext mit anderen Personen austauschen möchte. Jörg Knappens Abneigung gegen die Verwendung von 8-Bit-Zeichen im \LaTeX -Quelltext gipfelt in Empfehlungen, mit Werkzeugen wie `sed` oder `recode` zu arbeiten; das vorgegebene `sed`-Skript ersetzt dabei `ß` durch `\"s`, was als »š« im Druckbild erscheint. Es scheint deswegen wenig wahrscheinlich, dass der Autor selbst zu solchen Techniken greift.

Im gleichen Kapitel wird die Definition eigener Befehle und Umgebungen besprochen, was aus meiner Sicht zumindest für einen Anfänger etwas zu knapp ausfällt. Es wird auch gezeigt, wie man selbstständig eine Listenumgebung mit `list` und den entsprechenden Parametern erstellen kann. Ein zusätzlicher Hinweis auf `paralist` wäre hier sicherlich sinnvoll gewesen, weil es bereits zusätzliche Listen zur Verfügung stellt. Sehr gut finde ich den Abschnitt »Die goldenen Regeln des Makroschreibens« auf Seite 133 und der sich anschließende Abschnitt, der noch mal das Konzept der logischen Auszeichnung betont.

Leider wird in einem Beispiel in diesem Abschnitt der veraltete Schalter `\tt` anstatt `\ttfamily` verwendet. [1, 4]

Kapitel 12 beschäftigt sich auf zwölf Seiten mit PDF und HTML, wobei gerade PDF auf dreieinhalb Seiten für meinen Geschmack zu kurz kommt, obwohl der Buchrücken eine ausführliche Beschreibung der »Neuentwicklungen pdfT_εX und L^AT_εX2html« ankündigt. Bei der Verwendung von `latex2html` wird nicht erwähnt, welche Probleme auftreten können, wenn andere als die Standardklassen verwendet werden; `latex2html` kann auch nichts mit den Paketen `ngerman` und `fontenc` anfangen.³ Diese Probleme hat `tex4ht` nicht, das wird aber nicht erwähnt. In Bezug auf pdfT_εX ist zu lesen, dass `pstricks` und `eepic` damit nicht funktionieren, `pstricks` aber durch `pdftricks` ersetzt werde. Andere Lösungen, um doch `pstricks` direkt mit pdfT_εX verwenden zu können, werden nicht genannt. Der Leser erfährt auch hier nichts von den zusätzlichen Grafikformaten (png, jpeg), die mit pdfT_εX verwendet werden können. Wenigstens wird das Paket `hyperref` genannt, das einem PDF-Dokument automatisch vollwertige Hypertext-Funktionalität für Querverweise und Verzeichnisse verleiht.

Kapitel 13 erklärt die teilweise kryptischen Fehlermeldungen von T_εX und wo ihre Ursachen liegen, was insbesondere für Anfänger ausgesprochen hilfreich ist.

Zu jedem Text gehört auch das Korrekturlesen, was der Autor in **Kapitel 14** bespricht. Er beschränkt sich dabei nicht nur auf Vollständigkeit und Orthografie, sondern geht auch auf Gestaltung und Satz ein: »2. Korrektur des Absatzumbruchs (Trennungen, überlange Zeilen etc.) 3. Korrektur des Seitenumbruchs (Schusterjungen und Hurenkinder, Platzierung der Tabellen und Bilder etc.)«. Zur Rechtschreibprüfung wird auf das UNIX-Tool `aspell` verwiesen. Im Zusammenhang mit Silbentrennung wird darauf hingewiesen, dass innerhalb einer `\cite`-Anweisung nicht umbrochen werden kann (Seite 167). Leider wird nicht auf das Paket `cite` hingewiesen, das dieses Problem behebt. Ansonsten erhält der Leser in diesem Kapitel hilfreiche Hinweise, wie er Probleme, die der automatische Satz von T_εX erzeugt, lösen kann.

Das **letzte Kapitel** beschäftigt sich mit der Zukunft von L^AT_εX und geht in vier Sätzen noch mal auf pdfT_εX und `hyperref` ein.

Es gibt noch einen kurzen Abschnitt »Grafische Oberflächen für L^AT_εX«, in dem undifferenziert L_YX, das veraltete K_LY_X und Scientific Workplace

³Version 2002-2-1 (1.71), die laut `latex2html`-Homepage die aktuellste ist.

erwähnt werden, wobei in teilweise fehlerhafter Weise nicht zwischen den Produkten differenziert wird. Wirkliche grafische Oberflächen (statt den hier genannten Textverarbeitungssystemen mit L^AT_EX-Backend) wie Kile oder T_EXnicCenter, oder gängige Editoren und spezielle L^AT_EX-Editierlösungen werden nicht erwähnt. vim (samt latex-suite) wird nicht erwähnt, WinEdt auch nicht. Editieren als Bestandteil der Dokumenterstellung bleibt komplett außen vor: Die T_EX-Shell-Funktionalität beschert an unerwarteter Stelle (»Tippfehler in Befehlen«) Emacs und AUC_TE_X noch durch einen Halbsatz einen Kurzauftritt in Buch und Abspann, äh Stichwortverzeichnis, was anderen Editoren und Shells verwehrt bleibt.

Der **Anhang** beginnt mit »Informationsquellen zu L^AT_EX«. Dort wird T_EXLive als Distribution der Wahl genannt, weil sie von »zwei Dutzend T_EX-Anwendervereinigungen gemeinsam herausgegeben« wird und »installationsfertige T_EX-Systeme für alle wichtigen Betriebssysteme (Linux, MS Windows, Apple Mac OS X, Free BSD und weitere UNIX-Varianten)« enthält. Es wird auch darauf hingewiesen, dass Linux-Distributionen in der Regel bereits mit »T_EX und L^AT_EX 2_ε mit einer großen Auswahl an Paketen« ausgeliefert werden. Der Leser wird dann noch auf CTAN und DANTE e.V. (einschließlich *Die T_EXnische Komödie*) aufmerksam gemacht. Schließlich werden dem Leser als Anlaufstelle bei Problemen auch noch die Mailingliste TEX-D-L, die Newsgroup `de.comp.text.tex` und die DE-T_EX-FAQ [2] genannt. Somit ist er über die wichtigen Informationsquellen informiert.

Anhang B beschäftigt sich mit L^AT_EX und den Besonderheiten der deutschen Sprache. Hier wird der Leser auch noch mal auf `ngerman` hingewiesen, was wichtig ist, weil in den Beispielen davor immer `german` verwendet wird. In den dann noch folgenden vier Anhängen findet der Leser noch eine Übersicht mathematischer Symbole und erfährt sehr viel Nützliches über »Dateien und ihre Endungen«, »Zeichensätze« und »Die europäischen Schriften«.

Fazit

In der Newsgroup `de.comp.text.tex` entstand die Idee, die vorhandene L^AT_EX-Literatur nach geeigneten Büchern für Anfänger zu durchsuchen. In diesem Kontext wurde auch das vorliegende Buch rezensiert. »Schnell ans Ziel mit L^AT_EX 2_ε« würde ich keinem Anfänger empfehlen. Ich konnte keinen roten Faden entdecken und die Informationsdichte setzt zuviel L^AT_EX- und Computer-Kenntnisse voraus. Außerdem werden an manchen Stellen Fehler oder Probleme provoziert, die in der Newsgroup `de.comp.text.tex` immer wieder auftau-

chen und zum Teil auch schon in der FAQ [2] besprochen werden oder auch in `l2tabu` [4] zu finden sind. Nicht zu vergessen: Es wird eine funktionierende \TeX -Installation vorausgesetzt.

Davon abgesehen finde ich die Leserführung nicht sehr gelungen. Dies hängt einerseits mit der Gliederung und der bereits genannten Gestaltung zusammen, andererseits fehlen immer mal wieder Verweise auf Abbildungen, die dann unvermittelt auftauchen, oder es werden Formulierungen verwendet, die auch mir Kopfzerbrechen bereiteten. Die häufigsten Notizen lauteten bei mir »versteht kein Anfänger«, »zu knapp« und »too much«.

Negativ bewerte ich auch die Tatsache, dass die Bibliografie- und Indexerstellung und die Querverweise zu kurz kommen. Die zwei erstgenannten Bereiche werden auf insgesamt nur vier Seiten, die Verwendung von Querverweisen in Abschnitt 6.1 in einem fünf Zeilen umfassenden Satz abgehandelt.⁴ Die Ansicht des Autors, `BIB \TeX` lohne sich für Diplom- oder Doktorarbeiten nicht, teile ich nicht. Zumal einem bei Verwendung von `BIB \TeX` zahlreiche hilfreiche Frontends zur Eingabe der Literatur zur Verfügung stehen und man mit diesen auch seine stetig anwachsende Literatur unabhängig von `BIB \TeX` komfortabel verwalten kann. Allein deshalb lohnt sich meines Erachtens schon der Einsatz von `BIB \TeX` bzw. eines entsprechenden Frontends. Es wäre außerdem in einem Buch, dem es um wissenschaftliche Dokumente geht, sicherlich auch angemessen gewesen, auf Lösungen zur Erstellung von Präsentationen mit \LaTeX einzugehen. Dass KOMA-Script nicht besprochen wird, finde ich auch bedauerlich, da es mit seiner einheitlichen Schnittstelle für viele Einstellungen dem \LaTeX -Anwender – und gerade dem Anfänger – das Leben leichter macht.

Alles in allem entstand der Eindruck, dass ein technisch versierter \TeX -Profi einen Teil seiner Arbeitsweise dokumentiert, die teilweise nicht mehr auf dem neuesten Stand ist, und keine klare Zielgruppe im Blick hat. Es ist ein Buch, bei dem ich nicht weiß, wem ich es empfehlen sollte, auch wenn es unbestritten gute Kapitel und Erklärungen enthält.

Jörg Knappen. Schnell ans Ziel mit $\LaTeX 2_{\epsilon}$. 2., völlig überarbeitete Auflage. Oldenbourg Verlag. 244 Seiten. 2004. ISBN: 3-486-27447-3. Broschur, 29,80 €.

⁴Wobei der Satzspiegel etwa 10 cm x 19,5 cm beträgt bei einer Schriftgröße von 11pt, was zu etwa 60 Zeichen pro Zeile führt.

Literatur

- [1] Frank Mittelbach und Michel Goossens: *The L^AT_EX Companion*; Addison Wesley; 2. Aufl.; 2004.
- [2] Bernd Raichle, Rolf Niepraschk und Thomas Hafner: *Fragen und Antworten (FAQ) über das Textsatzsystem TeX und DANTE, Deutschsprachige Anwendervereinigung TeX e. V.*; 2003; <http://www.dante.de/faq/de-tex-faq/>.
- [3] Axel Reichert: *Satz von Tabellen*; 1999; <ftp://ftp.dante.de/tex-archive/info/german/tabsatz/tabsatz.ps>.
- [4] Mark Trettin: *Das L^AT_EX 2_ε-Sündenregister oder Veraltete Befehle, Pakete und andere Fehler*; 2004; <ftp://ftp.dante.de/tex-archive/info/12tabu/german/12tabu.pdf>.

»Der L^AT_EX-Wegweiser« von Christine Detig

Christian Faulhammer, Frank Küster

Nach vielen Jahren hat die beliebte Einführung eine Überarbeitung erfahren. Das Buch bietet für Anfänger kompakt alle grundlegenden Informationen, meistens gut erklärt und rund. Einige Schwächen haben sich dennoch eingeschlichen.

»Der L^AT_EX-Wegweiser« von *Christine Detig* [1] hat sich zum Ziel gesetzt, dem Leser die Grundkonzepte von L^AT_EX zu vermitteln und ihm zu helfen, »die vielen Möglichkeiten zu nutzen, die L^AT_EX zur Erstellung [...] befriedigender Dokumente bietet« – jedoch ohne Perfektionismus, d. h. ohne auf Feinheiten der Typografie einzugehen und unter Verzicht auf weiterführende Details oder die Erörterung von L^AT_EX-Internia. Die erste Auflage erschien bereits 1997 im selben Verlag und wurde in [3] besprochen. Die zweite stark überarbeitete Auflage erschien bereits 2004, deren Besprechung leider wegen Zeitmangels auf sich warten ließ.

Die Totale

Die Autorin, Christine Detig, ist keine Unbekannte im \TeX -Lager und auch bei DANTE e.V. Das lässt hoffen, dass das Buch seinen Anspruch einlösen kann. In der Tat macht es beim ersten Durchblättern einen guten Eindruck: Eine übersichtliche Typografie, die auch als \LaTeX -typisch erkennbar ist, dazu sehr viele Beispiele mit Code auf der linken Seite und der Ausgabe rechts daneben. Das Inhaltsverzeichnis verspricht auch, alle wesentlichen Themen abzuhandeln, die man für ein solides Grundlagenwissen braucht. Unglücklicherweise hat es die offizielle Internetseite¹ nicht mehr auf den Umschlag geschafft, dort werden gefundene Fehler und weiterführende Informationen dokumentiert.

Die Großaufnahme

Im ersten Kapitel werden Vor- (und Nach-)teile von \LaTeX , die Arbeitsweise im Vergleich zu WYSIWYG-Textverarbeitungen, die Beziehung von \TeX , \LaTeX usw. und das Konzept expliziten und strukturellen Markups vorgestellt. Zur Veranschaulichung folgt ein erstes Beispieldokument mit Ein- und Ausgabe und der Behandlung typischer (Tipp-)Fehler, und schließlich werden \TeX -Benutzeroberflächen vorgestellt mit Screenshots von \TeX nicCenter und AUCTION . Allgemein ist dieser Ansatz sehr Unix-lastig, sicherlich ist es für \LaTeX die beste Umgebung, aber die Realität sieht nun mal anders aus. Zudem kommt der Verweis auf den sehr guten Anhang B (u. a. Informationen zur Installation) etwas spät. Gold wert ist ein Abschnitt über typische Fehler und Fallen in \LaTeX und eine Interpretationshilfe für Fehlermeldungen.

Das Kapitel finden wir insgesamt gut gelungen, allerdings taucht hier auch das erste Mal eine Eigenheit auf, die uns im ganzen Buch gestört hat: Die Autorin verwendet für \LaTeX -Makros ausschließlich den Begriff »Tag«. Explizites Markup wird am Beispiel von HTML eingeführt, wo dieser Begriff gebräuchlich ist und wo er vielen Lesern begegnet sein wird. Wer aber in online erhältlichen \LaTeX -Anleitungen oder der Dokumentation zu einzelnen Paketen stöbert, wird ihm kaum begegnen. Ein Einsteigerbuch müsste die Leser daher auch mit den gebräuchlichen Begriffen »Makro« und »Kommando« vertraut machen. Der Autorin rutscht einige Male ein unnötiger Anglizismus heraus, dieser Umstand wurzelt wohl in ihrer Informatik-Herkunft.

¹<http://www.latex-wegweiser.de/>

Im zweiten Kapitel wird einerseits das Konzept der Dokumentklassen, Pakete und Optionen vorgestellt, andererseits wird erläutert, wie man einfache Dokumentstrukturen erhält – vor allem Titelei und Gliederung. In den folgenden Kapiteln werden dann nacheinander die häufig verwendeten Typen von Befehlen und Umgebungen abgehandelt. Durchgängig wird darin von den Standardklassen ausgegangen, um nicht für eine Aufgabenstellung mehrere Vorgehensweisen schildern zu müssen; die KOMA-Skript-Klassen werden leider nur gegen Ende kurz vorgestellt.

Im Kapitel drei geht es um Textstrukturen mit Listen, *verbatim*-Umgebungen und direkten Eingriff in den Satz mit Wechsel von Schriftart und -größe; hier wird auch auf Silbentrennung, manuellen Zeilenumbruch, Leerräume und die Anpassung des Satzspiegels mit *geometry* eingegangen. Das nächste Kapitel widmet sich auf 20 Seiten dem Tabellensatz, wobei auch *tabularx*, *longtable* und *colortbl* besprochen werden. Gerade hier gelingt es der Autorin gut, den formulierten Anspruch umzusetzen – Schritt für Schritt wird der Leser weitergeführt und ihm Lösungen für gebräuchliche Problemstellungen angeboten. Dass gewisse Tabellenelemente wie senkrechte Linien, typografisch umstritten sind, darauf wird zwar hingewiesen, der Wunsch vieler Autoren, sie trotzdem zu verwenden, wird aber ernst genommen. Lediglich die Besprechung von *tabular** hätten wir auf einen Halbsatz im Abschnitt zu *tabularx* beschränkt.

Kapitel fünf behandelt bewegliche Elemente, also Gleitumgebungen, Fuß- und Endnoten und Querverweise, das sechste Kapitel Grafiken, vor allem das *graphicx*-Paket. Bei den Gleitumgebungen wird das *caption2*-Paket als Nachfolger von *caption* angesehen. Seit Dezember 2003 ist es aber wieder genau umgekehrt (letzteres Paket mit der internen Versionsnummer 3.x), dieser Umstand scheint an der Autorin vorbeigelaufen zu sein. Bei solch einer unglücklichen Benennung ist das auch kein Wunder, selbst im T_EX-Catalogue² war dieser Umstand nicht verzeichnet. Ebenso ging es mit dem Paket *subfigure*, das Anfang 2004 von *subfig* abgelöst wurde. Die Erläuterungen zu Grafiken erklären die Eigenheiten von L^AT_EX ausführlich und zeigen Wege auf, wie man Grafiken relativ simpel aufbereitet.

Das siebente Kapitel dreht sich um Mathematiksatz, und es gelingt der Autorin, auf 20 Seiten einen guten Einstieg und eine Fülle von Beispielen einschließlich der *ams*-Pakete zu präsentieren, mit denen man sehr gut arbeiten kann. Eine vollständige Darstellung will man auch gar nicht bieten, die meis-

²<http://texcatalogue.sarovar.org/>

ten Fachgebiete brauchen nur relativ wenig speziellen Mathematiksatz, für den Rest existiert [4]. Grob falsch ist $\$$. . . \$\$$ als legitim für die Matheeingabe und $$. . .$$ dagegen als T_EX-Syntax zu bezeichnen; es ist genau umgekehrt. Richtig schön dagegen die Formatierung der Code-Schnipsel, die von Anfang an eine übersichtliche Einrückung für Mathematik präsentieren.

Im achten Kapitel werden Theoreme, Indexerstellung und Bibliografien mit B_IB_TE_X behandelt, letzteres erfreulich ausführlich. Die Auswahl an Zusatzpaketen und Informationen zu Zitierweisen ist wohlüberlegt und deckt viele Fälle ab; die Autor-Jahr-Zitierweise wird zumindest erwähnt, eine ausführlichere Darstellung findet man in [2]. Bei den Theoremen ist auch der einzige wirklich schwerwiegende Fehler zu finden: Das Beispiel ist nicht funktionsfähig, da ein falsches Paket geladen wird. Ebenfalls kompakt und dennoch vollständig (im weitesten Sinne) ist der Index-Teil, aber nicht unerwartet, wenn man weiß, dass die Autorin und Joachim Schrod am Index-Kapitel von [2] mitgewirkt haben.

Darauf folgt Kapitel neun über »besondere Dokumente«, nämlich Präsentationen und Folien, Briefe und die KOMA-Skript-Klassen. Dass erst in diesem Kapitel auch die Besonderheiten deutscher Dokumente mit (n)german, Trennhilfen und dem Euro-Symbol besprochen werden, halten wir für ungeschickt – zumal bereits im dritten Kapitel die Silbentrennung grundsätzlich behandelt wird. Immerhin gibt es dort schon einen Verweis auf die Beschreibung von (n)german. Die Beschränkung auf eine Präsentationsklasse (namentlich beamer) tut dem Buch gut, dass mehr davon existieren, wird dennoch erwähnt. Leider hat sich die Bedienung der Klasse in einigen wesentlichen Punkten seit Erscheinen des Buches geändert.

Das letzte Kapitel (unter dem etwas unglücklichen Namen »Postscript, PDF & Web«) behandelt die Erzeugung von PDF-Dateien, hyperref und die Verwendung von PSNFSS-Schriften, bevor in den Anhängen CTAN, Distributionen und Informationsquellen im Web vorgestellt werden. In diesen Abschnitten fehlt etwas der rote Faden, sie enthalten zusammenhanglos »alles was wir auch noch schreiben wollten«: Hier findet sich der Befehl zum Setzen des gesamten Dokuments in serifenloser Schrift, das include-Makro, die Option draft und eine knappe Erläuterung zu texdoc (siehe unten). Dass nicht nur auf CTAN, sondern auch auf DANTE e.V. und T_EX Users Group, die DE-T_EX-FAQ, Newsgroups und die Mailingliste tex-d-1 hingewiesen wird, ist jedoch lobend hervorzuheben. Auch die Liste mathematischer Symbole ist

nützlich und hier im Anhang richtig – griffbereit – platziert. Es folgt noch ein Index, mit dem man sich gut zurechtfindet.

Abspann

Den Kern des Buches machen die Kapitel drei bis zehn aus, und wie im Vorwort versprochen, werden die dort besprochenen Punkte in angemessener Knappheit dargestellt, ohne oberflächlich zu sein. Durch die vielen Beispiele sind die Erklärungen auch anschaulich, vor allem, weil sie immer gleichartig aufgebaut sind und man sich nicht jedes Mal neu hineindenken muss. Allerdings fiel einige Male negativ auf, dass die Beispiele nicht nur das im Text gesagte veranschaulichen, sondern gleich noch einige neue Aspekte einführen, die im Text nicht vorkommen: Das wird Anfänger sicher eher verwirren als zum Weiterdenken anregen. Lieber hätte man sich auch hier beschränkt; der Leser sollte ohnehin darauf gefasst sein, dass er sich im Einzelfall an anderen Orten weiter informieren muss. Allerdings wäre es gerade deswegen wichtig gewesen, die Leser regelmäßig auf andere Informationsquellen hinzuweisen. Verweise auf die Paketdokumentation finden sich leider äußerst selten, und die Frage, wie man sie findet, wird nur ganz knapp im Anhang unter »Informationen« zwischen Newsgroups und der FAQ erläutert (bedauerlicherweise fehlt auch noch ein Indexeintrag zu diesem Abschnitt). Darin wird im Wesentlichen auf `texdoc` und `texdoctk` verwiesen, aber der falsche Eindruck erweckt, beide deckten die gleichen Dokumente ab. Außerdem fehlt der Hinweis, dass `texdoc` auf der Basis von Dateinamen sucht – und daher z. B. mit den Argumenten »`babel`« oder »`hyperef`« zunächst nur die Catalogue-Einträge liefert, weil die Dateien `user.dvi` bzw. `manual.pdf` heißen und weil mehrere Dateien für Benutzer relevant sind. Stattdessen wird geraten, mit `kpsewhich -format='TeX system sources'` nach der `dtx`-Datei zu suchen. Dabei liegt die in der Regel. sowieso nur dann vor, wenn man ein Paket selber von CTAN geholt hat und liegt dann oft nicht im `texmf`-Baum. Wie man aus der `dtx`-Datei die Dokumentation erzeugt wird aber gar nicht erklärt, auch nicht bei der Beschreibung der Installation neuer Pakete von CTAN.

Getrübt wird das Gesamtbild dadurch, dass der Text einige dicke Fehler und Inkonsistenzen enthält. Das enttäuscht gerade bei einem Buch, das in der zweiten Auflage erscheint. Außerdem ist an mehreren Stellen die Reihenfolge nicht durchdacht, z. B. werden die Sternformen der Gliederungsbefehle (`section*` usw.) verwendet, bevor sie erklärt wurden (dasselbe gilt für die `gather`-Umgebung im Mathematikteil). Für die Zielgruppe kaum problema-

tisch, für die Rezensenten aber umso ärgerlicher waren die Stellen, an denen Computer Modern Roman als der Times Roman »recht ähnliche« Schrift eingeführt oder behauptet wird, D. E. Knuth habe das Copyright an die American Mathematical Society abgegeben. Erheiternd ist der Hinweis auf die käuflich zu erwerbende Schrift CM Bright, die man angeblich von »Microsoft Press« erhalten kann.

Die positiven Seiten sind aber überwiegend, so sind die Zusammenfassungen am Ende eines Kapitels sehr hilfreich zum Nachschlagen, und immer wieder geht das Buch in ein Frage-und-Antwort-Spiel über, in dem konkrete Probleme direkt angesprochen werden und viele nützliche Tipps enthalten sind. Damit bekommt man eine für diese Preiskategorie überdurchschnittliche Leistung.

Insgesamt ist das Buch eine gut zu verwendende Einführung in das Schreiben mit \LaTeX und eines der empfehlenswerten Bücher am Markt. Es ist ihm aber zu wünschen, dass es möglichst bald eine dritte Auflage erfährt, in der die vielen Fehler und Inkonsistenzen korrigiert werden. Die Besprechung der ersten Auflage ist nicht sonderlich ausführlich und zudem liegt eine sehr lange Zeit zwischen den beiden Auflagen, so dass ein Vergleich nicht möglich ist.

Literatur

- [1] Christine Detig: *Der \LaTeX -Wegweiser*; Mitp-Verlag; 2. Aufl.; Juli 2004; 268 Seiten, 14,95 €, ISBN 3-8266-1414-3, <http://www.latex-wegweiser.de/>.
- [2] Michel Goossens, Frank Mittelbach et al.: *Der \LaTeX -Begleiter, m. CD-ROM*; Pearson Studium; 2. Aufl.; Okt. 2005; ISBN 3-8273-7166-X, <http://www.latex-project.org/guides/books.html>.
- [3] Thomas Koch: *Der \LaTeX -Wegweiser; Die \TeX nische Komödie*; 1/97, S. 65–66; Mai 1997.
- [4] Herbert Voß: *Math mode*; Okt. 2005; CTAN://info/math/voss/mathmode/.

Spielplan

Termine

- 8. 3.–10. 3. 2006** DANTE 2006 Frühjahrstagung und
34. Mitgliederversammlung
Freie Universität Berlin
<http://www.dante.de/dante2006/>
- 28. 4.–2. 5. 2006** The 14th GUST Conference and Annual Meeting
BachoT_EX 2006
Bachotek (Brodnica Lake District), Polen
<http://www.gust.org.pl/BachoTeX/>
- 5. 7.– 8. 7. 2006** 16th EuroT_EX Conference
»A Hungarian T_EX Rhapsody«
Debrecen, Ungarn
<http://www.matexhu.org/eurotex2006>
- 9. 11.–11. 11. 2006** TUG 2006
Marrakesch, Marokko
<http://tug.org/tug2006/>

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.: 030/3481316
rolf.niepraschk@ptb.de
Gasthaus Pali-Eck
Koppenstr. 41
Zweiter Donnerstag im Monat, 19.00 Uhr

Bremen

Martin Schröder
Tel.: 0421/2239425
martin@oneiros.de
Wechselnder Ort
Erster Donnerstag im Monat, 18.30 Uhr

Darmstadt

Karlheinz Geyer
karlheinz.geyer@LHSystems.com
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. 5a (Ecke Bergerallee)
40213 Düsseldorf
Zweiter Mittwoch in ungeraden Monaten,
20.00 Uhr

Erlangen

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

Freiburg

Heiko Oberdiek
Tel.: 0761/43405
oberdiek@uni-freiburg.de
Wechselnder Ort
Dritter Donnerstag 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.: 06221/544527
luzia.dietsche@urz.uni-heidelberg.de
China-Restaurant »Palast«
Lessingstraße 36
Letzter Mittwoch im Monat, 20.00 Uhr

Karlsruhe

Klaus Braune
Tel.: 0721/6084031
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://zaubberer.net/latexwiki>

Köln

Helmut Siegert
Institut für Kristallographie
Zülpicher Straße 49b
Letzter Mittwoch im Monat, 19.30 Uhr

München

Michael Niedermair
 m.g.n@gmx.de
Wirtshaus »Löwe am Markt«
Dreifaltigkeitsplatz 4
Erster Dienstag 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

Wuppertal

Andreas Schrell
 Tel.: 02 02/50 63 81
schrell@wupperonline.de
Restaurant Croatia »Haus Johannisberg«
Südstraße 10
an der Schwimmoper Wuppertal-Elberfeld
Zweiter Donnerstag im Monat, 19.30 Uhr

Zürich

Johannes Reese
 reese@spw.unizh.ch
nach Vereinbarung

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:	Volker RW Schaa	president@dante.de
Vizepräsident:	Klaus Höppner	vice-president@dante.de
Schatzmeister:	Tobias Sterzl	treasurer@dante.de
Schriftführer:	Günter Partosch	secretary@dante.de
Beisitzer:	Thomas Koch	
	Bernd Raichle	advisor@dante.de

Server

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

Autoren/Organisatoren

- Christoph Bier** [48] **Frank Küster** [57]
 Kahnstraße 5
 34233 Fulda
 christoph.bier@web.de
 Sandstr. 5
 CH-8003 Zürich
 frank@kuesterei.ch
- Christian Faulhammer** [57] **Herbert Möller** [25]
 Lochnerstr. 26
 52064 Aachen
 christian@faulhammer.org
 Elsternweg 10
 48329 Havixbeck
 mollerh@math.uni-muenster.de
- Jürgen Fenn** [44] **Volker RW Schaa** [4]
 Friedensallee 174/20
 63263 Neu-Isenburg
 juergen.fenn@GMX.DE
 siehe Seite 66
- Ulrike Fischer** [19] **Martin Schröder** [47]
 Kastanienstraße 2
 53721 Siegburg
 martin@oneiros.de
- Klaus Höppner** [4] **Wolfgang Stieler** [6]
 siehe Seite 66
 Heise Verlag
 48] Helsdorferstraße 7
 30625 Hannover
 wst@heise.de
- David Kastrup** [48] **Herbert Voß** [3]
 Kriemhildstr. 15
 44793 Bochum
 dak@gnu.org
 Wasgenstr. 21
 14129 Berlin
 hvoss@tug.org
- Markus Kohm** [11]
 Fichtenstraße 63
 68535 Edingen-Neckarhausen
 markus.kohm@gmx.de

Die T_EXnische Komödie

18. Jahrgang Heft 1/2006 Februar 2006

Impressum

Editorial

Hinter der Bühne

4 Grußwort

T_EX-Theatertage

6 Freude, die ein Maler empfindet

Bretter, die die Welt bedeuten

11 Farbig hinterlegte Kopfzeilen mit KOMA-Script

19 Trennhilfen

25 Die GaPFil-Methode zur Erzeugung von L^AT_EX-picture-Umgebungen

T_EX-Beiprogramm

44 Neue Pakete auf CTAN

47 Berichtigungen zu »T_EX und die Druckvorstufe«

Rezensionen

48 »Schnell ans Ziel mit L^AT_EX 2_ε« von Jörg Knappen (2. Auflage)

57 »Der L^AT_EX-Wegweiser« von Christine Detig

Spielplan

63 Termine

64 Stammtische

Adressen

67 Autoren/Organisatoren