

Die T_EXnische Komödie

DANTE
Deutschsprachige
Anwendervereinigung T_EX e.V.

12. Jahrgang Heft 3/2000 September 2000

3/2000

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 an untenstehende Anschrift geschickt werden (entweder per E-Mail oder auf Diskette). Sind spezielle Makros, L^AT_EX-Pakete oder Schriften dafür nötig, so müssen auch diese mitgeliefert werden. Außerdem müssen sie auf Anfrage Interessierten zugänglich gemacht werden.

Diese Ausgabe wurde mit Hilfe folgender Programme fertiggestellt: **TeX**, Version 3.14159 (**Web2c 7.3.2x**), **LaTeX2e** <2000/06/01>, **windvi 0.67** (für Bildschirmdarstellung) und **dvips(k) 5.86d** (für Korrektur und Belichtung). Die Schriften zur Belichtung wurden mit dem METAFONT-Modus **linoone** (1270 dpi) berechnet.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 2300

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

E-Mail: dante@dante.de
dtk-redaktion@dante.de (Redaktion)

Druck: Konrad Tritsch Print und digitale Medien GmbH
Johannes-Gutenberg-Straße 1–3, 97199 Ochsenfurt-Hohe Stadt

Redaktion: Gerd Neugebauer (verantwortlicher Redakteur)
Luzia Dietsche Rudolf Herrmann Rolf Niepraschk
Günter Partosch Bernd Raichle Volker RW Schaa
Karin Schwind Peter Willadt

Redaktionsschluss für Heft 4/2000: 21. Oktober 2000

ISSN 1434-5897

Die T_EXnische Komödie 3/2000

Editorial

Liebe Leserinnen und Leser,

eigentlich müsste ich ja gar nichts mehr schreiben. In fast jedem Editorial steht im Grunde dasselbe. Ich rufe dazu auf, dass neue Beiträge für „Die T_EXnische Komödie“ eingereicht werden. Und immer wieder werden auch einige Beiträge geschrieben.

Trotzdem ist die Situation nicht erfreulich. Manchmal dauert es schon recht lange, bis eine Ausgabe voll ist und manchmal gibt es auch nur eine Ausgabe mit weniger Seiten, als ich mir wünschen würde.

In dieser Situation stellt sich mir die Frage, ob es eigentlich zu T_EX und Freunden so wenig zu sagen gibt. Oder ist es unsere Vereinszeitung, die so unattraktiv für Autoren ist? Wie steht es auf der anderen Seite mit den Lesern? Gelegentlich bekommen wir in der Redaktion eine Rückmeldung von einem Mitglied. Aber das ist eher die Ausnahme. In der Regel machen wir einfach die nächste Ausgabe in der Hoffnung, dass es da draußen doch noch einen Leser gibt, den das Ergebnis unserer Bemühungen auch interessiert.

Eine Art von Rückmeldung, die immer einmal wieder kommt, ist die Aufforderung, etwas zu einem bestimmten Thema abzudrucken. So etwas tun wir gerne, sofern sich jemand findet, einen entsprechenden Beitrag zu schreiben. Das ist aber eher selten.

Jeder der sich mit einem Aspekt von (L^A)T_EX befasst hat, müsste im Prinzip in der Lage sein, Einsichten zu vermitteln, die anderen helfen können. Damit wäre auch geklärt, wer denn nun solche Beiträge verfassen könnte – jeder von uns, der sich für seine Problemstellung eine Lösung erarbeitet! Es wäre also schön, wenn sich der Eine oder die Andere dazu durchringen könnte, doch etwas für „Die T_EXnische Komödie“ und unsere Leser zu schreiben.

Mit T_EXnischen Grüßen

Ihr Gerd Neugebauer

Hinter der Bühne

Vereinsinternes

Grußwort

Liebe Mitglieder,

wir hatten in die letzten Ausgabe von „Die T_EXnische Komödie“ die Ankündigung des $\mathcal{N}\mathcal{T}\mathcal{S}$ -Teams aufgenommen, dass die Java-Portierung von T_EX das Alpha-Teststadium erreicht hat. Nun können wir nach unserem Besuch bei der TUG 2000 in Oxford erfreut über die Vorführung der $\mathcal{N}\mathcal{T}\mathcal{S}$ -Version 0 (oder besser des *Old Typesetting Systems*) durch Karel Skoupý berichten. Als Beleg für das Erreichte führte er vor, wie $\mathcal{N}\mathcal{T}\mathcal{S}$ 0 das T_EXbook übersetzte. Auch wenn die aktuelle Version beträchtlich langsamer und noch nicht 100% (es war von 99,8% die Rede) kompatibel zu T_EX ist, so waren doch alle Anwesenden von dem Stand der Dinge beeindruckt. Im Anschluss fanden sich viele, die Interesse an einer Weiterentwicklung des Projektes haben. Auch die Finanzierung wurde durch die Zusicherung eines Betrages von 5 000 \$ durch die TUG entspannt.

Wir hoffen, dass das in Oxford von Entwicklern gezeigte Interesse zu einer schnellen Umsetzung und Förderung des Projektes führt. In den kommenden Wochen wird Karel Skoupý Bereinigungen am Code vornehmen und eine Basisdokumentation erstellen. Er hat zugesichert, zur Herbstveranstaltung in Hagen im Oktober einen weiteren Statusbericht mit Vorführung zu geben. Eine Freigabe des Codes wird erfolgen, wenn er alle erforderlichen Tests gemacht hat und sicher ist, dass das gesetzte Ziel erreicht wurde.

Damit bleibt uns, Sie alle zur Herbstveranstaltung und Mitgliederversammlung im Herbst 2000 vom 6.-7. Oktober nach Hagen/Westfalen einzuladen. Die Tagesordnung fanden Sie bereits in der letzten Ausgabe von „Die T_EXnische Komödie“.

An dieser Stelle möchten wir die Gelegenheit nutzen und Ihnen einen kleinen Überblick über bevorstehende und geplante T_EX-Tagungen zu geben.

Die Frühjahrstagung 2001 wird Anfang März in Rosenheim stattfinden, der genaue Termin wird sobald wie möglich bekanntgegeben. Wir danken schon jetzt Herrn Dr. Georg Lachenmayr für die Bereitschaft, die Tagung auszurichten.

Die TUG-Tagung 2001 unter dem Thema „ \TeX 2001 – A \TeX Odyssey“ findet wieder in den USA statt und zwar vom 12.–15. August 2001 an der Universität von Newark/Delaware (rund 50 km südwestlich von Philadelphia). Dadurch wird es im Spätsommer des nächsten Jahres auch wieder eine Euro \TeX -Tagung geben. Sie wird vom 23.–27. September 2001 in Kerkrade/Niederlande stattfinden. Es ist bereits jetzt möglich, sich unter <http://www.ntg.nl/eurotex> anzumelden. Für Anmeldungen bis Jahresende locken die Veranstalter mit] einem sehr günstigen Preis!

Und nun noch einen Ausblick auf die ferne Zukunft: Um die Ausrichtung der TUG2002 bewerben sich derzeit

- die TUGIndia für Kerala/Südindien (das Silicon Valley Indiens),
- Kairo/Ägypten und
- Cork/Irland.

Eine Entscheidung wird sicherlich erst im nächsten Jahr fallen; bis dahin wollen die Bewerber Einzelheiten über mögliche Kosten für Flug und Unterbringung liefern, damit die Konferenz von möglichst vielen \TeX ies besucht werden kann.

Mit freundlichen Grüßen

Thomas Koch Volker RW Schaa
(Präsident) (Vizepräsident)

Bretter, die die Welt bedeuten

Schöner Lesen – gut lesbare und optisch ansprechende Texte durch gute Typografie

Sven Storbeck

Jeder Autor eines Textes möchte, dass dieser auch gelesen wird. Zur Erfüllung dieses Wunsches tragen nicht nur die stilistischen Höhenflüge des Verfassers bei, sondern auch das äußere Erscheinungsbild des Textes. Der folgende Artikel – der sich an den typografisch Unbedarften wendet – zeigt, wie es dem Leser leicht gemacht werden kann, den Text zu erfassen.

Einleitung

Die Typografie eines Druckwerkes legt dessen Erscheinungsbild fest. Dazu gehören beispielsweise die Wahl des Papierformates, der Seitenränder, der Schriftart und -größe und des Zeilenabstandes. Hierzu gibt es eine Vielzahl von Tipps und verbindlichen Regeln, die oft schon so alt sind wie die Buchdruckerkunst selbst. Die Beachtung dieser Regeln stellt sicher, dass ein Text ermüdungsfrei gelesen werden kann und auch ästhetischen Ansprüchen genügt.

Mit der Typografie verhält es sich wie mit der Rechtschreibung oder auch wie mit Dreck: Gibt es keine Rechtschreibfehler oder ist alles sauber, fällt das keinem Menschen auf. Sowie aber ein Text von Fehlern wimmelt oder eine Küche von Schimmelkulturen überwuchert ist, erregt dies Anstoß. Wer sich die Mühe macht, einen Text typografisch einwandfrei zu setzen, wird also kaum zu hören bekommen: „Ui, war das *schön* zu lesen!“ Wer jedoch typografischen Murks produziert, macht es dem Leser schwer, den Inhalt zu erfassen.

ABCDEFghijkl Äöü ß 012345 &!?	normal
<i>ABCDEFghijkl Äöü ß 012345 &!?</i>	kursiv
ABCDEFghijkl Äöü ß 012345 &!?	fett
ABCDEF <i>GH</i> ijkl ÄÖÜ SS 012345 &!?	Kapitälchen

Abbildung 1: Schriftschnitte der „Computer Modern Roman“ (EC-Schriften)

Dieser Artikel beschäftigt sich mit der Gestaltung von Texten, deren einzige Aufgabe es ist, Informationen zu vermitteln. Solche Texte – wie auch dieser Text – werden gelesen, weil sich der Leser von vornherein für den Inhalt interessiert. Sie müssen nicht wie eine Werbeanzeige oder ein Flugblatt beständig „Lies mich! Bitte, bitte, lies mich!“ schreien. Für solche Texte gelten andere Regeln. Hier geht es um die Gestaltung von Texten, die vor allem wissenschaftlichen oder literarischen Charakter haben: wissenschaftliche und technische Veröffentlichungen, Studien-, Diplom- und Examensarbeiten, aber auch Prosa und Poesie. Zum Setzen solcher Texte ist \LaTeX hervorragend geeignet.

Dieser Artikel beschreibt, wie Text im deutschsprachigen Raum gesetzt wird. Andere Länder, andere Sitten: Im Ausland wird oft mehr oder weniger stark von den hier dargestellten Regeln abgewichen, beispielsweise werden Anführungs- und Gedankenstriche dort oftmals anders gesetzt.

Von der Schrift

Schriftarten und -familien

Jede Schrift ist ein Individuum. Sie hat ihr eigenes Erscheinungsbild, ihren eigenen Charakter, ihren eigenen Platzbedarf (jede Schrift „läuft“ unterschiedlich), ihre eigene Geschichte. Und sie hat Familie. Zu jeder Schrift gehören mehrere „Töchter“, zum Beispiel die elegante Kursive, die Fette (in jeder Familie gibt es schwarze Schafe), die Halbfette und die Kapitälchen. Diese Töchter werden *Schriftschnitte* genannt. Die Schriftschnitte der Standard- \LaTeX -Schrift können Abbildung 1 entnommen werden; die Verwendungszwecke der unterschiedlichen Schriftschnitte werden weiter unten behandelt.

Schriften lassen sich grob in zwei Familien aufteilen (siehe Abbildung 2). Die *Antiqua* ist der konservativere Zweig der Familie. An den Enden der

ABCDEFghijkl Äöü ß 012345 &!?	Antiqua
ABCDEFghijkl Äöü ß 012345 &!?	Grotesk

Abbildung 2: Antiqua und Grotesk

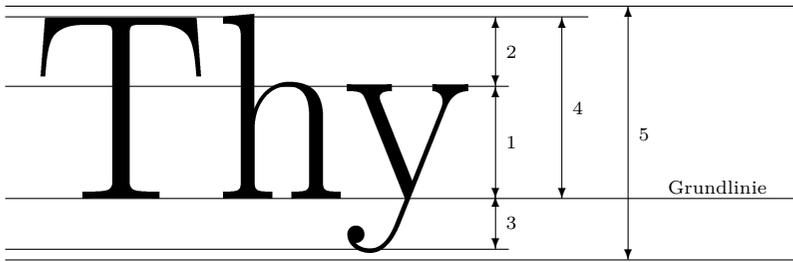
Buchstaben befinden sich kleine Querstriche, die *Serifen*. Dementsprechend wird eine Antiqua auch als *Serifenschrift* bezeichnet. Später entstand die *serifenlose Schrift*, die anfangs als so seltsam empfunden wurde, dass man sie *Grotesk* taufte.

Die Unterteilung der Schriften in Antiqua und Grotesk ist äußerst grob. Es gibt noch weitere Familien (Schreibschrift, gebrochene Schrift, serifenbetonte Linear-Antiqua) und Unterfamilien (Renaissance-Antiqua, Barock-Antiqua, klassizistische Antiqua usw.). Eine vollständige Abhandlung findet sich beispielsweise in [?].

Für längere Texte sollte eine Antiqua verwendet werden, sie kann von den meisten Menschen ermüdungsfreier gelesen werden. Die meisten Tageszeitungen, Romane und Fachbücher sind in diesen Schriften gesetzt. Die Grotesk eignet sich vor allem für kürzere Publikationen, Gebrauchsanleitungen, die Werbung oder große Überschriften.

Den Anwendern von Textverarbeitungsprogrammen steht heute eine riesige Anzahl von Schriften zur Verfügung. So mancher Laie zeigt sich von dieser Vielfalt derartig begeistert, dass er in einem Text so viele Schriften verwendet wie nur irgend möglich. Die Folge ist, dass der Text unruhig und unübersichtlich wird – einfach hässlich. Beim Verwenden mehrerer Schriften sind eine Menge Dinge zu berücksichtigen; die Schriften müssen miteinander harmonieren. Das Mischen von Schriften sollte man lieber dem Profi überlassen. Benutzen Sie so wenig unterschiedliche Schriften wie möglich!

Eine weitere Möglichkeit des Computers (wenn auch nicht unbedingt von \LaTeX), auf die sich der Laie mit Begeisterung stürzt, ist das Verfremden von Schrift. Text kann beliebig gedehnt, verbogen und mit allerlei Schnickschnack versehen werden. Betrachten Sie eine Schrift als ein Kunstwerk. Wer ein altes Gemälde verfremdet oder die Cover-Version eines bekannten Musikstückes aufnimmt, muss schon verdammt gut sein. Das sind die meisten Laien einfach nicht – ich möchte mich da nicht ausschließen.



Mittellänge (1), Oberlänge (2), Unterlänge (3), Versalhöhe (4), Schriftgröße (5)

Abbildung 3: Lettermaße

Schriftgröße

Die *Schriftgröße*, auch *Schriftgrad* genannt, wird in Punkten (abgekürzt pt) gemessen. Von wo bis wo wird aber nun die Schriftgröße gemessen? Abbildung 3 mag der Verdeutlichung dienen. Alle Buchstaben einer Zeile sind an der *Grundlinie* ausgerichtet. Dies gilt sowohl für die Großbuchstaben (*Versalien* oder *Majuskeln* genannt) als auch für die Kleinbuchstaben (*Gemeine* oder *Minuskeln* genannt). Die Höhe der meisten Gemeinen (,a', ,c', ,e', ,i', ,m', ,n', ,o', ,r', ,s', ,u', ,v', ,x', ,z') bezeichnet man als *Mittellänge* oder *x-Höhe*. In L^AT_EX entspricht die Mittellänge des jeweils aktiven Zeichensatzes der Länge „ex“.

Manche Gemeine weisen zusätzlich *Oberlängen* (,b', ,d', ,f', ,h', ,k', ,l', ,t') oder *Untерlängen* (,g', ,j', ,p', ,q', ,y') auf. Die *Versalhöhe* gibt die Höhe der Versalien an. In Abbildung 3 entspricht die Versalhöhe der Summe aus Mittellänge und Oberlänge; es gibt jedoch auch Schriften, bei denen die Versalhöhe größer oder kleiner sein kann. Versalien haben im Allgemeinen keine Unterlängen, eine Ausnahme bildet in den meisten Schriften das ,Q'. Damit die Buchstaben optisch die gleiche Höhe aufweisen, ragen Rundungen oftmals über die Begrenzungen hinaus, wie die Unterlänge beim ,y' in Abbildung 3.

Nun wäre es nahe liegend anzunehmen, dass die Schriftgröße der Summe aus Mittellänge, Oberlänge und Unterlänge entspricht – tatsächlich aber ist dem nicht so. Die Schriftgröße liegt leicht über diesem Wert; hinzu kommt nämlich der vertikale Abstand, der zwischen den Zeilen eingefügt wird, wenn mit einfachem Zeilenabstand gesetzt wird. Dieser Abstand variiert allerdings von Schrift zu Schrift.

ff fi fl ffi ffl & ß

Abbildung 4: Ligaturen in L^AT_EX

Als wenn das nicht schon kompliziert genug wäre, stellt sich zusätzlich die Frage, wie die Maßeinheit „Punkt“ definiert ist. Während sich in nahezu allen Bereichen das metrische System durchgesetzt hat, lebt die Typografie diesbezüglich noch immer im finstersten Mittelalter. Um es kurz zu machen: In L^AT_EX entspricht ein Punkt (pt) 0,351 46 mm. Wenn Sie es irgendwie vermeiden können, schlagen Sie sich nicht mit irgendwelchen anderen Maßeinheiten herum.

Die Schrift, aus der der eigentliche Text eines Dokumentes gesetzt ist, wird als *Grundschrift* oder *Brottschrift* bezeichnet. Hierfür eignen sich Schriftgrößen von etwa 9 bis 12 pt. Die Schriftgröße 12 pt entspricht in etwa der Schriftgröße einer Schreibmaschine. Die Schriftgrößen von 5 bis 8 pt bezeichnet man als *Konsultationsgrößen*. Aus diesen Größen werden Textteile gesetzt, die der Leser nur in Ausnahmefällen „konsultiert“, ein Beispiel dafür sind Fußnoten. Schriftgrößen über 12 pt werden *Schaugrößen* genannt, sie werden für Überschriften verwendet.

Ligaturen

Eine *Ligatur* ist die Verschmelzung von zwei oder drei Buchstaben zu einer Kombination. Eine typische Ligatur ist zum Beispiel das Verschmelzen von fi zu fi (siehe Abbildung 4).

Kaum ein DTP-Programm ist in der Lage, Ligaturen zu setzen. L^AT_EX bildet eine rühmliche Ausnahme, was für den Setzer allerdings streng genommen zusätzlichen Aufwand bedeutet. Ligaturen dürfen *nicht* gesetzt werden, wenn die Buchstabenkombination zwei verschiedenen Silben angehört: „Auflage“ statt „Auflage“. (Die Aufhebung der Ligatur erfolgt durch den Befehl `"|` oder alternativ durch `\/, also Auf"|lage.) Zudem sind die von LATEX verwendeten Ligaturen ffi und ffl im Deutschen nie benutzt worden. Wer also Ligaturen wirklich richtig setzen möchte, der muss den gesamten Text nach den möglichen Ligaturen (ff, fi, fl, ffi, ffl) durchsuchen und individuell entscheiden, ob eine Ligatur gefragt ist oder nicht. Ob dieser Aufwand gerechtfertigt ist, muss jeder für sich entscheiden... Weitere Regeln zum Setzen von Ligaturen können [?] entnommen werden.`

Andere berühmte Ligaturen sind ‚&‘ und ‚ß‘. Erstere ist aus den Buchstaben ‚e‘ und ‚t‘ vom lateinischen ‚et‘ entstanden (noch erkennbar in der Kursiven: ‚℘‘), letzteres aus den beiden Varianten des ‚s‘ in den gebrochenen Schriften.

Von der Seite

Papierformat

Bei der Wahl des Papierformates hat der europäische L^AT_EX-Anwender nur in seltenen Fällen die Qual der Wahl; im Allgemeinen wird das Format DIN A4 verwendet. Papier dieser Größe ist an fast jeder Straßenecke erhältlich, im Übrigen verfügt kaum jemand über einen Drucker oder Fotokopierer, der andere Formate verarbeiten kann. Diese äußeren Zwänge lassen in den meisten Fällen kaum ein anderes Format zu. Daher soll an dieser Stelle nicht diskutiert werden, ob die DIN-Serie mit ihrem Seitenverhältnis von 1 zu $\sqrt{2}$ nun wirklich der Ästhetik letzter Schluss ist. Wer die Möglichkeit hat, andere Formate zu verarbeiten, sei auf [?] verwiesen; hier wird ausgiebig auf dieses Thema eingegangen.

Satzspiegel

Der Aufbau der Seiten (Seitenränder, Kolummentitel, Paginierung)¹ wird *Satzspiegel* genannt. Der obere Rand wird als *Kopfstege* bezeichnet, der untere als *Fußstege*. Der linke Rand bei ungeraden Seitenzahlen und der rechte Rand bei geraden Seitenzahlen heißt *Bundstege*, der gegenüberliegende *Außenstege*. Werden die Seiten nur einseitig bedruckt, ist der Bundstege links und der Außenstege rechts.

Über den Satzspiegel – insbesondere die Breite der Ränder – ist schon viel geschrieben worden [?, ?]. Es gibt eine Vielzahl von Regeln, die einen besonders ästhetischen Satzspiegel erzeugen sollen. Eine Methode, nach der schon im Mittelalter Satzspiegel erstellt worden sind, zeigt Abbildung 5.

Durch die diagonalen Hilfslinien werden die Position und das Seitenverhältnis des Satzspiegels festgelegt. Die Größe ist frei wählbar, ausgehend von einem Wert können die übrigen drei errechnet werden. Wird der Kopfstege k vorgegeben, können der Außenstege a , der Bundstege b und der Fußstege f durch

¹ Eine Erläuterung dieser Begriffe erfolgt weiter unten.

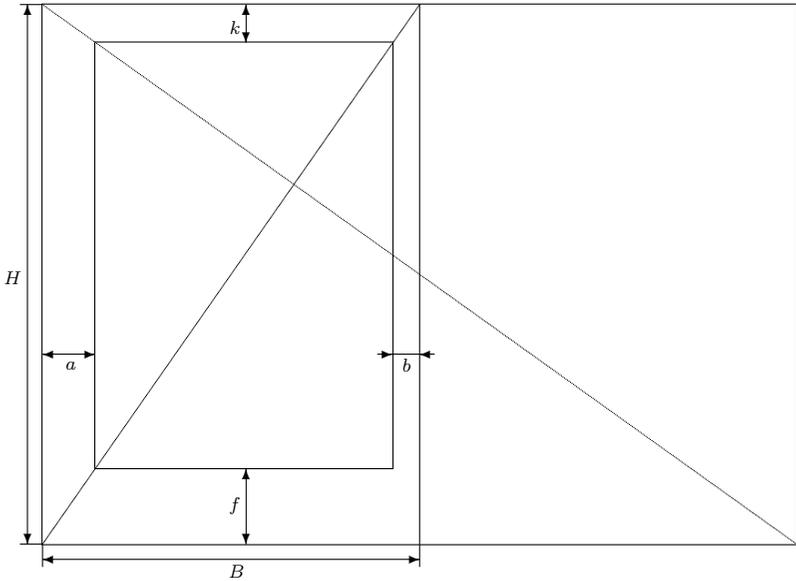


Abbildung 5: Konstruktion eines klassischen Satzspiegels, hier für eine gerade Seite

Anwendung der folgenden Gleichungen bestimmt werden (unter Verwendung der Seitenhöhe H und der Seitenbreite B):

$$a = \frac{2B}{H} k \quad (1)$$

$$b = \frac{B}{H} k \quad (2)$$

$$f = 2k \quad (3)$$

Bundsteg und Außensteg sowie Kopfsteg und Fußsteg stehen damit im Verhältnis eins zu zwei zueinander, der Satzspiegel hat das gleiche Seitenverhältnis wie die Seite. Die Höhe des Satzspiegels beträgt demzufolge $H - 3k$, die Breite $B(H - 3k)/H$.

Diese Konstruktion eines Satzspiegels eignet sich vor allem für literarische Werke wie Romane und Gedichtbände. Für wissenschaftliche und technische

Texte erscheint vor allem der Fußsteg als zu breit. Hier sollte die Funktion der Ästhetik übergeordnet werden. Beachten Sie dabei die folgenden Punkte:

- Achten Sie darauf, dass die Seitenränder nicht zu klein sind. Zu schmale Ränder konfrontieren den Leser mit einer sehr großen Menge Text pro Seite, was abschreckend wirkt. Außerdem haben schmale Ränder lange Zeilen zur Folge, die sehr schlecht gelesen werden können.
- Der Bundsteg sollte der schmalste sein, es folgen der Kopfsteg, der Außensteg und der Fußsteg.
- Drucken Sie eine Seite aus, betrachten Sie das Ergebnis aus einiger Entfernung und beurteilen Sie selber, ob das Ganze einigermaßen ansprechend aussieht.

Für eine DIN-A4-Seite können beispielsweise die folgenden Ränder gewählt werden: Bundsteg 10 mm, Außensteg 20 mm, Kopfsteg 15 mm, Fußsteg 25 mm. Je kleiner die verwendete Schrift ist, desto schmaler muss der Satzspiegel sein; eine gut lesbare Seite enthält maximal 55–60 Zeichen pro Zeile (inklusive Leerzeichen). Das Minimum liegt bei etwa 35 Zeichen pro Zeile.

Für die Wahl des Bundsteges muss die spätere Bindung berücksichtigt werden. Wird das Schriftstück in eine Mappe geheftet oder mit einer Klemmschiene zusammengehalten, ist nicht mehr die komplette Seite sichtbar. In einem solchen Fall muss der Bundsteg entsprechend breiter sein.

Der Satzspiegel ist eine Begrenzung, über die *nichts* hinausragen darf, auch keine Abbildungen oder Tabellen. Oben im Satzspiegel kann als Kopfzeile ein *lebender Kolumnentitel* gesetzt werden, d. h. die jeweilige Kapitelüberschrift erscheint oben auf der Seite. Im lebenden Kolumnentitel sollte nur die Überschrift der ersten Gliederungsebene (nicht die des jeweiligen Unterabschnittes) erscheinen. Das Verwenden von lebenden Kolumnentiteln bedingt, dass die Überschriften nicht zu lang sind, sonst passen sie nicht in die (stets einzeilige) Kopfzeile. Zwischen Kolumnentitel und Text muss ausreichend vertikaler Zwischenraum gelassen werden, zur deutlicheren Trennung kann auch eine horizontale Linie gesetzt werden.

Beim Setzen der Seitenzahlen spricht man von *Paginierung*. Die Seitenzahl steht entweder ausnahmsweise außerhalb des Satzspiegels unten auf der Seite (zentriert, davor und dahinter oft jeweils ein Gedankenstrich und ein Leerzeichen: „– 42 –“) oder außen im lebenden Kolumnentitel. Das Wort „Seite“ wird der Seitenzahl nicht vorangestellt.

Grauwert einer Seite

Ein gut gesetzter Text wirkt aus einiger Entfernung betrachtet wie eine gleichmäßige ruhige, graue Fläche. Dies gilt allerdings nur für Sachbücher, wissenschaftliche und technische Veröffentlichungen und Romane. Bei einem Text, der beständig „Komm, lies mich!“ rufen soll, wie in Illustrierten oder in der Werbung, wirkt eine solche „Bleiwüste“ abschreckend. Daher werden solche Texte mit Abbildungen, Text in Kästchen usw. garniert.

Gestört wird dieser *Grauwert* zum Beispiel durch Überschriften. Das ist auch so gewollt – der Leser soll die Überschrift schließlich auf Anhieb finden können. Schlecht ist die Störung des Grauwertes zum Beispiel durch unnötige fette Auszeichnungen, Sperrungen, Verwenden mehrerer nicht harmonisierender Schriften oder zu große Wortabstände, die durch stümperhaften Blocksatz entstehen.

Zum Prüfen des Grauwertes bedient sich jeder seiner eigenen Methode. Wichtig ist, dass man den Text nicht lesen kann, sonst konzentriert sich das Auge auf einzelne Wörter. Manche halten die Seite vor einen Spiegel, andere betrachten die Seite von hinten gegen das Licht oder kneifen die Augen so weit zusammen, dass Lesen nicht mehr möglich ist. Ich persönlich schiele einfach über den Rand meiner Brille. Auch aus extremer Kurzsichtigkeit lassen sich Vorteile ziehen.

Von Abständen und Absätzen

Zeichen-, Wort- und Zeilenabstand

Ein wichtiges Maß für Abstände in der Typografie ist das *Geviert*. Dabei handelt es sich um ein Quadrat, dessen Kantenlänge der verwendeten Schriftgröße entspricht: ■. Mit L^AT_EX kann durch den Befehl `\quad` ein horizontaler Leerraum von der Breite eines Gevierts erzeugt werden, die zugehörige Maßeinheit heißt „em“. Die Befehle `\quad` und `\hspace{1em}` entsprechen einander.

Der Abstand zwischen den Buchstaben wird durch die verwendete Schrift festgelegt. Jede Schrift – und auch jeder Schriftschnitt – hat eine eigene *Laufweite*; deutlich wird dies aus den Abbildungen 1 und 2. Im Allgemeinen empfiehlt es sich nicht, an der Laufweite herumzuspielen. Nur beim Setzen von Versalien sollte die Laufweite leicht erhöht werden, man spricht vom *Sperren* (s. u.). Für jede Schrift ist festgelegt, wie dicht einzelne Buchstabenkombinationen aneinander gesetzt werden. Bei bestimmten kritischen Kombinationen

wird *unterschnitten*, d.h. die Zeichen werden so dicht aneinander gesetzt, dass nicht zu viel weißer Raum entsteht (,VA‘ statt ‚VA‘). L^AT_EX erledigt das im Allgemeinen automatisch.

Der Wortabstand ist beim links- oder rechtsbündigen Satz (*Flattersatz*) zwischen allen Wörtern gleich groß. Er beträgt bei der von L^AT_EX als Standard verwendeten Computer Modern etwa ein Drittelgeviert. Ein schmales Leerzeichen, das durch `\`, erzeugt wird, hat eine Breite von einem Sechstelgeviert, entspricht also einem halben Leerzeichen. Kritisch kann der Wortabstand werden, wenn Blocksatz verwendet wird – wie in diesem Text. Im nächsten Abschnitt wird näher darauf eingegangen.

Um den Zeichenabstand muss man sich also nicht viel kümmern, um den Wortabstand nur, wenn L^AT_EX entsprechende Meldungen ausgibt. Anders sieht es mit dem Zeilenabstand aus. Hier lassen sich kaum Regeln aufstellen, deren Beachtung automatisch zu einem guten Ergebnis führen. Ein gut gewählter Zeilenabstand hängt von Schriftart, Laufweite, Schriftgröße, Zeilenlänge und Wortabstand ab. Als Faustformel gilt: Je länger die Zeile, desto größer der Zeilenabstand. Wird der Zeilenabstand zu klein gewählt, findet der Leser beim „Zurückspringen“ des Auges vom Ende einer Zeile den Anfang der nächsten Zeile nicht. Ist der Zeilenabstand zu groß, wirkt der Text zerrissen.

Wird der Text mit einem erhöhten Zeilenabstand gesetzt (in Diplom- und Examensarbeiten wird beispielsweise zumeist ein anderthalbfacher Zeilenabstand verlangt), sollten die Überschriften davon abweichend mit normalem Abstand gesetzt werden. Außerdem sollten Überschriften nicht im Blocksatz sondern linksbündig im Flattersatz gesetzt werden. Das macht L^AT_EX nicht freiwillig. Um die gewünschte Formatierung zu erreichen, müssen die Befehle zur Erzeugung der Überschriften neu definiert werden; hier wird stellvertretend für die `section`-Ebene gezeigt, wie es funktioniert:

```
\makeatletter
\renewcommand\section{\@startsection {section}{1}{\z@}%
{-3.5ex \@plus -1ex \@minus -.2ex}%
{2.3ex \@plus .2ex}%
{\renewcommand{\baselinestretch}{1}\flushleft%
\normalfont\Large\bfseries}}
\makeatother
```

Die entsprechenden Befehle können aus der jeweiligen `cls`-Datei (in diesem Fall `report.cls`) kopiert, um die unterstrichenen Änderungen erweitert und in die Präambel der Datei aufgenommen werden. Dieses Verfahren muss für alle verwendeten Ebenen durchgeführt werden. Auch Fußnoten, Bild-, Tabellenunterschriften u. ä. sollten mit einfachem Zeilenabstand gesetzt werden, in [?] wird erklärt, wie die Befehle umdefiniert werden können.

Blocksatz oder Flattersatz?

Im *Flattersatz* (meist linksbündig, selten rechtsbündig) hat jede Zeile die Länge, die sie „von Natur aus“ benötigt. Die Abstände zwischen den Wörtern sind alle genau gleich groß; die Schrift ist in ihrem Element und darf sich genau so entfalten, wie sie möchte. Das hört sich erst einmal nicht schlecht an, die Schrift darf so sein, wie es ihrem Naturell entspricht. Doch wo Licht ist, ist auch Schatten: Der Preis für den (linksbündigen) Flattersatz ist, dass der (rechte) Rand äußerst unruhig wirkt – ein Beispiel dafür ist dieser Absatz.

Blocksatz „flattert“ nicht; die rechte Kante des Textes ist ebenso gerade wie die linke. Die Seite wirkt dadurch wesentlich ruhiger. Der Blocksatz wird dadurch erzeugt, dass die Wortabstände so variiert werden, dass jede Zeile gleich lang ist, man spricht vom *Ausschließen*. Zu Zeiten des Bleisatzes war das eine äußerst aufwändige Angelegenheit; heute dagegen kann auch das schlechteste Textverarbeitungsprogramm Block setzen. Allerdings: Ganz so einfach ist es nun auch wieder nicht. Oftmals muss von Hand eingegriffen werden, um einen guten Blocksatz zu erzeugen.

In bestimmten Situationen (besonders wenn das Satzprogramm einzelne Wörter nicht trennen kann), werden die Wortabstände so groß, dass Löcher in den Text gerissen werden (siehe Abbildung 6). \LaTeX nörgelt in solchen Situation mit den Meldungen `Overfull \hbox` oder `Underfull \hbox`. Oftmals kann diesem Übel abgeholfen werden, indem man \LaTeX zeigt, wo eine Silbentrennung möglich ist; bei Verwendung der Schreibmaschinenschrift (`\texttt{...}`) weigert sich \LaTeX generell zu trennen. Andernfalls kommt man meist durch ein leichtes Umformulieren des Textes zum Ziel („nie“ statt „niemals“ schreiben o. ä.). Je größer die Schrift und je kleiner die Zeilenlänge, desto größer wird die Wahrscheinlichkeit, dass es Probleme mit dem Wortabstand gibt.

Jeweils eine Drehstrom-Asynchronmaschine mit einer Nennleistung von 2,6 MW dient zum Antrieb der Kesselspeisepumpen.

Abbildung 6: Gepfuschter Blocksatz – die Wortabstände sind viel zu groß

Flattersatz hat dort seine Berechtigung, wo die Zeilen so kurz sind, dass ein hochwertiger Blocksatz praktisch nicht mehr möglich ist: bei mehrspaltigem Satz, in Marginalien, Tabellen, Überschriften und Literaturverzeichnissen. Außerdem sollte beachtet werden, dass ein guter Flattersatz (die Länge der Zeilen weicht nicht zu sehr voneinander ab; dies wird auch *Rauhsatz* genannt) ebenso viel Aufwand bedeutet wie der Blocksatz.

Absatzuntergliederung

Wird ein neuer Absatz begonnen, so muss dies deutlich erkennbar sein. Dazu bieten sich zwei Möglichkeiten an: Die eine Möglichkeit besteht darin, die laufende Zeile zu beenden und eine neue Zeile zu beginnen, ohne dass zusätzlicher Leerraum zwischen den Absätzen eingefügt wird.

Ist die letzte Zeile eines Absatzes zufällig gerade so breit wie der Satzspiegel, wird der Absatzwechsel jedoch nicht deutlich. Daher wird die erste Zeile eines Absatzes eingezogen, d. h. am Anfang der ersten Zeile steht ein kleiner Leerraum (so wie in diesem Absatz). Die Länge des Absatzeinzuges sollte etwa der Schriftgröße (also einem Geviert) entsprechen. Nach Überschriften entfällt der Absatzeinzug, das macht L^AT_EX automatisch. Zur Frage, ob auch nach Aufzählungen, Gleichungen, Abbildungen usw. kein Absatzeinzug gesetzt wird, gibt es unterschiedliche Meinungen. Ich persönlich setze keinen Einzug, sofern es sich nicht um gleitende Objekte handelt. Mit `\setlength{\parindent}{1em}` in der Präambel wird der Absatzeinzug gesetzt.

Die andere Möglichkeit zur Absatzuntergliederung besteht darin, nach dem Absatz einen vertikalen Leerraum zu setzen, dieser beträgt meist eine halbe oder eine Leerzeile. Ein Absatzeinzug wird nicht gesetzt.

Der Nachteil dieser Methode besteht darin, dass mehr Platz benötigt wird und der Text leicht zerrissen wirkt. Als Beispiel dient – abgesehen vom vorletzten Absatz – dieser Artikel. (Diese Methode der Absatzuntergliederung wird von der Dokumentklasse für „Die T_EXnische Komödie“ vorgegeben.) Der Abstand zwischen den Absätzen wird durch eine Zuweisung an die Länge `\parskip` eingestellt. Dies sollte ein elastisches Maß sein [?].

Jeweils eine Drehstrom-Asynchronmaschine mit einer Nennleistung von 2,6 MW dient zum Antrieb der Kesselspeisepumpen. Die Nennspannung dieser

Abbildung 7: So nicht! Zahl und zugehörige Einheit dürfen nie umbrochen werden.

Ausrichten von Text

Zum Ausrichten von Text werden Tabulatoren oder Tabellen (ggf. ohne Umrahmung) verwendet. Niemals sollten Wörter untereinander gesetzt werden, indem so viele Leerzeichen hintereinander eingegeben werden, dass es so einigermaßen hinkommt. L^AT_EX wehrt sich dagegen ohnehin mit Händen und Füßen, so dass dem Anwender gar nichts anderes übrig bleibt, als mit Tabulatoren oder Tabellen zu arbeiten. Aber manchmal ist man ja auch auf andere Programme angewiesen. Leerzeichen sind flexible Zeichen, die vom Textverarbeitungsprogramm in bestimmten Grenzen gedehnt werden können. Zudem hat in einer Proportionschrift jeder Buchstabe seine eigene Breite. (Ein ‚W‘ ist deutlich breiter als ein ‚i‘.)

Kontrolle des Umbruchs

Zeilenumbruch

Ein sehr beliebter Fehler sind Zeilenumbrüche an der falschen Stelle: Abkürzungen, die aus mehreren Teilen bestehen, und Kombinationen von Zahlenwert und Einheit dürfen nie umbrochen werden (siehe Abbildung 7). Am einfachsten wird ein derartiger Umbruch verhindert, indem bereits bei der Eingabe des Textes ein geschütztes Leerzeichen verwendet wird. Dieses Sonderzeichen verhindert automatisch, dass an seiner Stelle ein Umbruch erfolgt. Die Funktion des geschützten Leerzeichens wird mit L^AT_EX durch die Tilde (~) erfüllt. Ein halbes geschütztes Leerzeichen wird durch \, erzeugt.

Seitenumbruch

Steht die letzte Zeile eines Absatzes am Anfang einer neuen Seite, so wird dies in der etwas rauen Fachsprache der Drucker als *Hurenkind* bezeichnet. Steht die erste Zeile eines Absatzes alleine am Ende einer Seite, während der Rest des Absatzes auf der folgenden Seite steht, wird dies *Schusterjunge* genannt. Ganz besonders hässlich ist das, wenn es sich bei dieser Zeile auch noch um eine Überschrift handelt.

Schusterjungen und Hurenkinder sind zu vermeiden. Es gibt mehrere Möglichkeiten, um dies zu erreichen: Der jeweilige Absatz wird komplett auf die andere Seite gebracht, der Text wird leicht gekürzt oder man schwadroniert ein bisschen was hinzu. (Eine weitere Möglichkeit besteht darin, *dezent* an den Einstellungen für Laufweite und Zeilenabstand des entsprechenden Absatzes zu drehen. So etwas wird gemeinhin als Pfusch bezeichnet. Extra-Pfui!²) \LaTeX versucht ohnehin, Schusterjungen und Hurenkinder zu vermeiden. Aber manchmal klappt es eben nicht.

Hilfreich für die Kontrolle des Seitenumbruches ist es, ein neues Kapitel (nur erste Gliederungsebene) immer mit einer neuen Seite zu beginnen. Andernfalls kann es vorkommen, dass das nachträgliche Einfügen eines einzelnen Wortes den mühsam erstellten Umbruch des gesamten Textes zerstört. Wird ein Kapitel stets mit einer neuen Seite begonnen, wird diese Gefahr minimiert (wenn auch nicht vermieden). In der `chapter`-Ebene wird von \LaTeX automatisch eine neue Seite begonnen, wenn ein neues Kapitel beginnt.

Silbentrennung

Die automatische Silbentrennung ist bei \LaTeX – im Gegensatz zu anderen Programmen – standardmäßig aktiviert. Das ist auch gut so, denn ohne diese Möglichkeit entsteht ein sehr löcheriger Blocksatz bzw. ein extrem flatternder Flattersatz.

Wird falsch getrennt, lassen sich durch `\-` die korrekten Trennstellen angeben. Durch den `hyphenation`-Befehl kann \LaTeX beigebracht werden, wie ein Wort im gesamten Text getrennt wird [?].

Dem ‚st‘ tut die Trennung nicht mehr weh: Silbentrennung für die neue Rechtschreibung lässt sich bei neueren \LaTeX -Installationen einstellen, indem in der Präambel statt `\usepackage{german}` der Befehl `\usepackage{ngerman}` verwendet wird. Näheres zur Silbentrennung von \LaTeX kann beispielsweise den FAQ-Seiten von DANTE e.V. (<http://www.dante.de/faq/de-tex-faq>) entnommen werden.

Sinnentstellende und verwirrende Trennungen müssen vermieden werden. Da \LaTeX zwar setzen aber nicht lesen kann, ist hier Handarbeit gefragt („Ur-instinkt“ statt „Urin-stinkt“). Einige Wörter erhalten durch falsche Trennungen völlig andere Bedeutungen: „Druck-erzeugnis“, „Drucker-zeugnis“.

² Aber wenn man es geschickt macht, merkt es kein Mensch. Man muss nur wirklich *sehr vorsichtig* sein.

Auszeichnungen

Zum Hervorheben eines Wortes oder einer Textpassage gibt es eine ganze Reihe von *Auszeichnungen*. Die wichtigsten werden in den folgenden Abschnitten vorgestellt. Prinzipiell kann zwischen zwei Arten von Auszeichnungen unterschieden werden:

- Ein Wort oder mehrere Wörter sollen im Text hervorgehoben werden, ohne dass der Leser beim Blick auf die Seite sofort mit der Nase darauf gestoßen wird. Erst beim Lesen fällt die Auszeichnung auf. Diese Art der Hervorhebung wird durch die Kursive und die Kapitälchen erfüllt.
- Der Leser soll die ausgezeichnete Passage auf Anhieb finden, zum Beispiel zum Auffinden eines Stichwortes oder einer Überschrift. Hierzu werden die übrigen Methoden der Auszeichnung angewendet, sie haben jedoch Einfluss auf den Grauwert der Seite.

Allgemein gilt, dass nicht mehr als eine Auszeichnung gleichzeitig verwendet werden sollte. Ein Wort wird nicht auffälliger, wenn es kursiv, fett und unterstrichen gesetzt wird – nur hässlicher. Außerdem sollten Auszeichnungen sparsam verwendet werden, ansonsten verlieren sie ihre Wirksamkeit. Zur Verdeutlichung: Während meines Studiums hatte ich einen Kommilitonen, der seine Skripte und Bücher praktisch komplett mit einem Textmarker anzustreichen pflegte. Viel genützt hat es ihm nicht.

Kursiv setzen

Um einzelne Stellen im laufenden Text besonders hervorzuheben, werden sie *kursiv* gesetzt. Diese Methode wird angewendet, wenn einem Wort besonderer Ausdruck verliehen werden soll. („Der Stromwandler darf *auf gar keinen Fall* im Leerlauf betrieben werden.“) Außerdem werden Wörter aus anderen Sprachen häufig kursiv gesetzt. („Dieses Verfahren ist *de jure* unzulässig.“) In *diesem* Text dient die Kursive zusätzlich zur Einführung neuer Begriffe. Die Kursive erzeugt nur eine leichte Unruhe im Textbild. Sie fällt beim Lesen zwar auf Anhieb auf, „schreit“ jedoch nicht aus dem Text heraus.

Soll ein Wort in einem kursiv gesetzten Satz hervorgehoben werden, wird dieses wieder normal gesetzt: „*Öffnen Sie die Abdeckung* niemals *im laufenden Betrieb!*“ Beim Übergang von kursiv auf normal ist es teilweise nötig, \LaTeX

durch den Befehl `\/` daran zu hindern, dass es den letzten kursiven Buchstaben in den nächsten normalen ragen lässt. Durch Verwenden des Befehls `\emph{}` wird dieser Ausgleich jedoch automatisch ausgeführt.

Übrigens: Die Kursive ist ein eigener Schriftschnitt, sie besteht nicht aus einer lediglich nach rechts geneigten senkrechten Schrift. Liegt für eine Schriftart keine Kursive vor, wird diese von vielen Programmen durch Neigen der senkrechten Schrift simuliert. Dem kann man recht schnell auf die Schliche kommen, wenn man sich das kleine ‚a‘ näher ansieht (kursiv ‚a‘, geneigt ‚a‘). \LaTeX kann beides: kursiv („Paragraf“, erzeugt durch `\emph{Paragraf}`) und geneigt („Paragraf“, erzeugt durch `\textsl{Paragraf}`).

Kapitälchen

Eine weitere Form der dezenten Auszeichnung sind die KAPITÄLCHEN. Sie werden verhältnismäßig selten verwendet, sind aber durchaus eine sehr schöne Form der Auszeichnung. Angewendet werden sie meistens, um die Namen von Personen oder Firmen anzugeben. („Nach GUTSCH ergibt sich daraus...“)

In Wörtern, die aus Kapitälchen oder Versalien (s. u.) gesetzt sind, wird niemals ‚ß‘ gesetzt („SCHLIESSEN“ statt „SCHLIEßEN“). \LaTeX erledigt das automatisch und wandelt ‚ß‘ in ‚ss‘ um.

\LaTeX verwendet – wie es sich gehört – „echte“ Kapitälchen, d. h. die Kapitälchen sind wie die Kursive ein eigener Schriftschnitt. Andere Textverarbeitungsprogramme verwenden oft eine verkleinerte Form der normalen Schrift, was sich dadurch bemerkbar macht, dass die Buchstaben dünner in Erscheinung treten als die der normalen Schrift (echt: „KAPITÄLCHEN“, falsch: „KAPITÄLCHEN“).

Fett setzen

Im Gegensatz zur Kursiven fällt ein **fett** gesetztes Wort sofort ins Auge. Diese Möglichkeit der Auszeichnung wird ausschließlich genutzt, wenn der betreffende Text auf Anhieb gefunden werden soll, zum Beispiel in Überschriften. Zum Bekräftigen eines bestimmten Wortes wird niemals fett gesetzt; der Grauwert der Seite würde dadurch erheblich beeinflusst werden. Die Kursive erfüllt diesen Zweck weitaus besser.

Versalien

Auch durch das Setzen aus VERSALIEN (Großbuchstaben) kann Text hervorgehoben werden. Wörter aus Versalien sollten leicht gesperrt werden (ungesperrt: „VERSALIEN“). Außerdem empfiehlt es sich beim Satz aus Versalien, um einen bis anderthalb Punkte kleiner zu setzen. Dies kann durch das Paket `soul` erledigt werden. Für eine 10-pt-Schrift stehen dafür die folgenden Befehle in der Präambel:

```
\usepackage{soul}
\newfont{\versalZehn}{ecrm10 at 9pt}
\capsdef{////10}{\versalZehn}{.07em}{.45em plus.1em minus.1em}%
{.45em plus.1em minus.1em}
```

Im Text kann nun durch `\caps{VERSALIEN}` das Wort „VERSALIEN“ gesetzt werden. Näheres zur Konfiguration kann der Dokumentation des `soul`-Paketes entnommen werden.

Achtung: Der \LaTeX -Luxus, dass ‚ß‘ in ‚SS‘ umgewandelt wird, funktioniert nur bei Kapitälchen, beim Setzen von Versalien muss man selber darauf achten.

Unterstreichungen

Das Setzen von Unterstreichungen ist in nahezu allen Fällen ein Anachronismus. Die Unterstreichung stammt aus der Zeit der Schreibmaschine, mit der andere Formen der Auszeichnung wie kursiv und fett nicht möglich waren. Immerhin streicht \LaTeX – im Gegensatz zu vielen anderen Textverarbeitungsprogrammen – die Unterlängen beim ‚g‘, ‚j‘, ‚p‘, ‚q‘ und ‚y‘ nicht kurzerhand *durch*, dafür gerät der Unterstrich aber bei kleinen Zeilenabständen in gefährliche Nähe zur nächsten Zeile.

Sperren

Eine weitere Form der Auszeichnung ist das S p e r r e n. Hier wird der Abstand zwischen den Buchstaben vergrößert. Vom Sperren sollte man ganz einfach die Finger lassen – es gibt schönere Formen der Auszeichnung. Die einzige Ausnahme besteht beim Setzen von Wörtern in Versalien.

Wechsel der Schriftart

Der Wechsel der Schriftart fällt oftmals kaum auf. Zudem ist das Mischen von Schriftarten eine kritische Angelegenheit. Für die Angabe von Computer-Befehlen, Pfadangaben, Internet-Adressen usw. wird oft auf eine **Schreibmaschinenschrift** (`LATEX: \texttt{}`) umgeschaltet, was allerdings großen Einfluss auf den Grauwert hat. („Die T_EXnische Komödie“ gibt für derartige Fälle den Wechsel auf `\texttt` allerdings fest vor, so dass diese Form der Auszeichnung auch in diesem Artikel Anwendung findet.)

Wechsel der Schriftfarbe

Der Wechsel der Schriftfarbe ist ein weiteres Mittel der Auszeichnung. Dies ist jedoch eine eher unübliche Methode und verbietet sich oftmals zudem aus technischen Gründen. Das Drucken mehrfarbiger Texte ist in kleinen und mittleren Auflagen noch immer sehr teuer.

Schreibregeln

Für viele Anordnungen in der Typografie können keine verbindlichen Regeln aufgestellt werden, hier sind Augenmaß und ästhetisches Empfinden gefragt. Anders verhält es sich mit den folgenden Schreibregeln. Sie sind quasi „Gesetz“ – wenn es auch in einigen Details unterschiedliche Auffassungen gibt.

Zahlensatz

Im laufenden Text werden Zahlen bis einschließlich zwölf ausgeschrieben, abgesehen von wenigen Ausnahmen („12 bis 14 Stunden“ statt „zwölf bis 14 Stunden“, „Seite 3“ statt „Seite drei“). In Tabellen werden ausschließlich Ziffern gesetzt. Bei einer Verbindung von Zahl und Einheit werden entweder Zahl *und* Einheit ausgeschrieben („drei Meter“) oder die Zahl aus Ziffern gesetzt und die Einheit abgekürzt („3 m“, niemals „3 Meter“ oder „drei m“).

Zwischen Zahl und Einheit wird ein halbes oder ein ganzes Leerzeichen gesetzt. (Die Meinungen und Argumente zum Thema „halbes oder ganzes Leerzeichen zwischen Zahl und Einheit“ sind vielfältig; suchen Sie sich was aus. In jedem Fall falsch ist *kein* Leerzeichen.) Um einen Zeilenumbruch zwischen Zahl und Einheit zu verhindern, sollte hier ein halbes Leerzeichen (`\,`) verwendet werden.

Werden Zahl und Einheit zum Bestandteil eines Substantives, werden zwischen Zahl und Einheit und zwischen Einheit und nachfolgendem Wort Bindestriche ohne Leerraum gesetzt („10-kV-Schaltanlage“, nicht „10 kV-Schaltanlage“). Ausnahme: „5%ige Lösung“.

Lange Zahlen sollten der besseren Übersichtlichkeit halber durch ein halbes Leerzeichen von rechts in Dreiergruppen untergliedert werden („100 000“). Auch Nachkommastellen werden in Dreiergruppen gegliedert: „3,141 592“. Darüber, ob auch bei vierstelligen Zahlen untergliedert wird, scheiden sich die Geister („1 345“, „1345“). Betrachten wir es als eine Kann-Regelung. Eine Untergliederung durch Punkte („1.453“) ist nicht zu empfehlen. (Im anglo-amerikanischen Raum wird der Punkt als Dezimaltrennzeichen verwendet.) Seitenzahlen, Jahreszahlen und Postleitzahlen werden nicht untergliedert.

Beim Setzen von Dezimalzahlen wird ein Komma – kein Punkt – verwendet („3,141 592“). Im mathematischen Modus fügt \LaTeX nach dem Komma horizontalen Leerraum ein („3, 141 592“), da \LaTeX das Komma nicht als Dezimaltrennzeichen interpretiert. Dies kann durch Eingabe von $\$3{,}14159\$$ umgangen werden. Eine elegantere Lösung findet sich auf den FAQ-Seiten von DANTE e.V.

Telefon-, Telefax- und Postfachnummern werden von rechts nach links in Zweiergruppen gegliedert, sofern die Nummer aus mehr als drei Ziffern besteht. Untergliedert wird durch ein halbes Leerzeichen („4 58 44 83“). Bei siebenstelligen Telefonnummern in der Schweiz werden die ersten drei Ziffern nicht untergliedert („922 71 31“). Die Vorwahl wird in Klammern der Rufnummer vorangestellt, Rufnummern von Nebenstellenanschlüssen werden durch einen Bindestrich ohne Leerraum abgetrennt: „(05 11) 4 30-16 16“. Alternativ wird die Vorwahl manchmal auch durch einen Schrägstrich von der Rufnummer abgetrennt: „05 11/4 30-16 16“.

Uhrzeiten werden meistens in der Form „14.30 Uhr“ gesetzt. Ebenfalls zulässig sind die Schreibweisen „14:30 Uhr“ und – ziemlich hässlich – „14³⁰ Uhr“.

Die von \LaTeX als Standard verwendeten Ziffern bezeichnet man als *Versalziffern*, sie haben alle die Höhe der Versalien („1234567890“). Zusätzlich gibt es die *Mediävalziffern*, die – wie die Gemeinen – auch Unterlängen haben („1234567890“). Im laufenden Text stören die Mediävalziffern den Grauwert nicht so stark wie die Versalziffern. In technischen Texten ist die Verwendung von Mediävalziffern jedoch nicht üblich, in Tabellen *dürfen* diese Zif-

fern nicht verwendet werden. In \LaTeX kann mit `\oldstylenums{3\,055}` die Zahl „3 055“ erzeugt werden.

Abkürzungen

Besteht eine Abkürzung aus mehreren Teilen, wird zwischen die einzelnen Teile ein *halbes* Leerzeichen gesetzt („z. B.“, „u. a.“, „z. T.“). \LaTeX kann das, viele andere Textverarbeitungsprogramme aber nicht. Wer ausnahmsweise einmal auf ein solches Programm angewiesen ist, sollte lieber ein ganzes Leerzeichen als gar keines setzen. Am Anfang eines Satzes wird nicht abgekürzt sondern ausgeschrieben. Zum Beispiel wird am Anfang dieses Satzes nicht „Z. B.“ gesetzt.

Abkürzungen, die den Fachjargon handlicher machen, sind nützlich („KZA“ statt „Kühlturnzusatzwasseraufbereitungsanlage“). Viele andere Abkürzungen können jedoch häufig vermieden werden. Im Fließtext sollte besser „Abbildung“ statt „Abb.“ geschrieben werden. Die Abkürzung „bzw.“ kann in den meisten Fällen durch „und“ oder „oder“ ersetzt werden. Nach der Abkürzung „d. h.“ steht *kein* Komma.

Steht eine Abkürzung am Satzende, wird nur *ein* Punkt gesetzt. Wird der Satz durch ein Ausrufezeichen oder Fragezeichen beendet, stehen Punkt *und* Ausrufezeichen oder Fragezeichen.

Klammern und Anführungszeichen

Diese Zeichen werden ohne Leerraum an die eingeschlossenen Wörter gesetzt: „(Der Einfluss des Skin-Effektes wird vernachlässigt.)“ statt „(Der Einfluss des Skin-Effektes wird vernachlässigt.)“

Als Anführungszeichen kann sowohl „Anführungszeichen“ als auch »Anführungszeichen« gesetzt werden. In der Schweiz werden diese Zeichen umgedreht («Anführungszeichen»). Im deutschsprachigen Raum nicht zulässig ist die englische Schreibweise “Anführungszeichen”.

Bindestrich

Das Setzen von Binde- und Gedankenstrichen belegt in der Hitliste der meist-begangenen typografischen Fehler einen der vorderen Plätze. Dabei ist es eigentlich ganz einfach: Der Bindestrich („-“, \LaTeX : -) wird nur zur Untergliederung von Wörtern und für die Silbentrennung verwendet. Er steht

ohne Leerraum zwischen den zu gliedernden Wortteilen („DIN-Norm“ statt „DIN - Norm“). Der Bindestrich ist kürzer als der Gedankenstrich und das Minuszeichen.

Gedankenstrich

Der Gedankenstrich („–“, \LaTeX : `--`) dient zur Untergliederung von Satzteilen und ist länger als der Bindestrich (so lang wie ein halbes Geviert). Vor und hinter dem Gedankenstrich steht ein Leerzeichen. („Der Strom – und damit auch die Verlustleistung – wird vernachlässigbar klein.“ statt „Der Strom -und damit auch die Verlustleistung- wird vernachlässigbar klein.“) Der Gedankenstrich sollte nie am Anfang einer Zeile stehen, durch die Verwendung von geschützten Leerzeichen kann dies erreicht werden: **Der Strom~-- und damit auch die Verlustleistung~-- wird vernachlässigbar klein.** Auch als Auslassungsstrich bei der Angabe von Währungen („54,- DM“), als Spiegelstrich bei Aufzählungen und zur Angabe „von-bis“ („30-35 A“, *ohne* Leerzeichen) findet der Gedankenstrich Verwendung, weiterhin als Streckenstrich („der ICE München-Hamburg“) und als Zeichen für „gegen“ im Sport („Werder Bremen – Bayern München“, diesmal wieder *mit* Leerzeichen).

Geviertstrich

Der Geviertstrich („—“, \LaTeX : `---`) findet im deutschen Satz kaum Verwendung. Er ist so lang, dass er das Schriftbild arg aufreißt. Einzig in Tabellen wird er benutzt. (Um anzuzeigen, dass eine Zelle keinen Eintrag hat.)

Minuszeichen

Auch das Minuszeichen („–“, \LaTeX : `$-$`) hat seine eigene Länge (so lang wie das Pluszeichen). Für die negative Zahl „-8“ sollte also **\$-8\$** geschrieben werden, nicht **-8** (erzeugt „-8“).

Schrägstrich

Zur Verwendung des Schrägstriches mag sich in der typografischen Literatur niemand so recht äußern. Fest steht, dass im Allgemeinen vor und hinter dem Schrägstrich kein Leerraum steht („Groß-/Kleinschreibung“). Es gibt jedoch Fälle, in denen das recht seltsam anmutet, zum Beispiel in

„Bereich Fertigung/Bereich Verwaltung“.

Eigentlich bilden die beiden ersten und die beiden letzten Wörter jeweils einen Begriff. Der Schrägstrich zieht jedoch die beiden mittleren Wörter zusammen, während das erste und das letzte etwas unmotiviert in der Gegend herumstehen. In diesem Fall empfehle ich, vor und hinter dem Schrägstrich ein halbes Leerzeichen zu setzen:

„Bereich Fertigung / Bereich Verwaltung“

Doppelpunkt

Der Doppelpunkt wird immer ohne Leerzeichen an das vorangegangene Wort gesetzt. (Eine Ausnahme besteht in Literaturverzeichnissen nach DIN, hier wird er als Trennzeichen mit einem Leerzeichen davor und dahinter verwendet.) Folgt nach dem Doppelpunkt ein ganzer Satz, wird groß weiter geschrieben, folgt kein ganzer Satz, wird klein weiter geschrieben. Beispiele: „Aus diesen Annahmen ergibt sich die folgende Definition: Der Fehler wird zu 1 % angenommen.“, „In diesem Raum sind die folgenden Anlagen installiert: ein Kompressor, ein Kältetrockner und ein Windkessel.“

Apostroph

Eigentlich eine Selbstverständlichkeit, dennoch einer der beliebtesten Fehler schlechthin: Im Deutschen sollte das besitzanzeigende ‚s‘ nie durch einen Apostroph abgetrennt werden.³ („Professor Blendermanns Skript“ statt „Professor Blendermann’s Skript“) Wer auch noch das ‚s‘ bei der Bildung des Plurals mit Apostroph abtrennt, gehört endgültig mit einem Duden erschlagen („CDs“, *nicht* „CD’s“). Die endgültige Katastrophe: „sonntag’s“ statt „sonntags“. Wer das schreibt, der schafft demnächst wahrscheinlich auch noch „Kek’s“.

Auslassungspunkte

Für die Auslassungspunkte (*Ellipsenzeichen*, „...“, `LATEX: \dots`) gibt es in jeder Schriftart ein eigenes Zeichen. In diesem Zeichen stehen die Punkte deutlich weiter auseinander als in drei einzeln getippten Punkten („...“). Die Auslassungspunkte werden mit einem kleinen Abstand an den vorangehenden Satz gehängt. Ich verwende dafür ein halbes Leerzeichen . . .

³ Okay: In [?] wird’s mittlerweile zähneknirschend gestattet. Aber schön ist es nicht!

Kaufmännisches Und-Zeichen

Das kaufmännische Und-Zeichen ‚&‘ wird *ausschließlich* in Firmennamen verwendet („Siemens & Halske“). „Obst & Gemüse“ sollten nur diejenigen über ihren Laden schreiben, die auch so heißen.

Ausrufezeichen und Fragezeichen

Ausrufe- und Fragezeichen werden ohne Leerraum an den vorangehenden Text angeschlossen. Steht vor dem Ausrufe- oder Fragezeichen eine Abkürzung, fällt deren Punkt *nicht* weg. („Achten Sie auf rotierende Maschinenteile, schwebende Lasten usw.!“)

Aufzählungen

Ist die Aufzählung Bestandteil eines Satzes, werden zwischen den Aufzählungspunkten Kommata gesetzt:

Das Ziel des Projektes ist es,

- die Strombezugskosten der Kunden zu senken,
- die Strombezugskosten der SWH zu senken und
- zum Schutz der Umwelt den CO₂-Ausstoß zu reduzieren.

Andernfalls werden keine Kommata gesetzt:

Das Projekt hat die folgenden Ziele:

- Senkung der Strombezugskosten des Kunden
- Senkung der Strombezugskosten der SWH
- Reduzierung des CO₂-Ausstoßes zum Schutz der Umwelt

Fußnoten

Bezieht sich eine Fußnote auf ein einzelnes Wort, steht das Fußnotenzeichen direkt hinter diesem Wort. Handelt es sich bei dem Wort um das letzte eines Satzes, steht es *vor* dem abschließenden Punkt. Bezieht sich die Fußnote auf den ganzen Satz, steht es *nach* dem Punkt. Beispiele: „Zur Durchführung der Messung dient ein Oszilloskop HP 3680 der Firma Hewlett-Packard⁴.“ Hier bezieht sich die Fußnote auf das Messgerät. „Zwischen Zahl und Einheit wird ein Leerzeichen gesetzt.⁵“ Hier bezieht sich die Fußnote auf den ganzen Satz.

Rechtschreibung

Kein Text ist frei von Tippfehlern; irgendetwas wird immer übersehen. Auch Sie werden in diesem Artikel möglicherweise noch den einen oder anderen Fehler gefunden haben. Diese Tatsache sollte jedoch keinesfalls dazu verleiten, nicht zu versuchen, möglichst wenig Fehler zu machen. Rechtschreibfehler sind einfach peinlich, sie hinterlassen einen denkbar schlechten Eindruck. Häufen sie sich, wird vom eigentlichen Inhalt des Textes abgelenkt. Deshalb sollte dieser möglichst von mehreren Personen Korrektur gelesen werden. Das ist ganz besonders wichtig, wenn man selber mit den Regeln der Rechtschreibung, Zeichensetzung und Grammatik etwas auf Kriegsfuß steht.

Zwei Anmerkungen seien noch zur neuen Rechtschreibung erlaubt: Wenn Sie die neue Rechtschreibung noch nicht beherrschen und nicht gezwungen sind, diese zu verwenden, verwenden Sie die alte. Niemals jedoch sollten neue und alte Rechtschreibung in einem Text gemischt werden. Wer „im allgemeinen“ (klein) schreibt, muss auch die alten Komma-Regeln verwenden!

Um einem weit verbreiteten Irrglauben abzuhelfen: Auch nach den Regeln der neuen Rechtschreibung ist das ‚ß‘ nicht gestorben. Es ist nur einfacher geworden zu entscheiden, ob ‚ß‘ oder ‚ss‘ geschrieben wird. Nach einem lang gesprochenen Vokal oder Diphthong⁶ wird ‚ß‘ gesetzt („Buße“), nach einem kurz gesprochenen Vokal oder Diphthong folgt ‚ss‘ („Busse“). Nach den Regeln der alten Rechtschreibung musste noch unterschieden werden, ob sich der ‚s‘-Laut am Ende eines Wortes oder einer Silbe befindet; dies ist nun nicht mehr

⁴Inventar-Nr. 12738-2

⁵Nach [?] wird ein halbes Leerzeichen gesetzt, wenn Zahl oder Einheit aus nur einem Zeichen bestehen.

⁶Ein Diphthong ist die Zusammensetzung aus zwei Vokalen oder einem Umlaut und einem Vokal: zum Beispiel ‚au‘, ‚eu‘, ‚äu‘.

erforderlich. „Strasse“ ist in jedem Fall falsch. (Ausnahme: die Schweiz. Dort gibt es schon lange kein ‚ß‘ mehr.)

Zusammenfassung

Wer die in diesem Artikel dargestellten Empfehlungen und Regeln konsequent anwendet, hat schon eine Menge geschafft: Er macht es seinem Leser leichter – auch, wenn der das gar nicht so recht gemerkt hat. Dies wird um so wichtiger, wenn das eigene Schicksal vom Leser des Textes abhängt, zum Beispiel bei Studien- oder Diplomarbeiten. Dennoch: Das Wesentlichste mag dargestellt worden sein, ein Fachbuch über Typografie kann dieser Artikel jedoch nicht ersetzen. Zum perfekten Typografen ist der Weg noch etwas länger – auch ich bin weit davon entfernt, diesen Titel für mich in Anspruch nehmen zu wollen. Kritik und Anmerkungen zu diesem Artikel nehme ich gerne unter der Mail-Adresse sven.storbeck@gmx.de entgegen.

Zeilenabstände

Peter Willadt

Der Zeilenabstand entscheidet mit darüber, wie gut ein Text lesbar ist. Zu kleine Zeilenabstände führen dazu, dass man die einzelnen Worte nur schwer entziffern kann; auch das Verrutschen des Auges von einer Zeile in eine andere wird dadurch begünstigt. Zu große Zeilenabstände sehen seltsam aus; statt eines gleichmäßigen Grauwertes wirkt die Seite „gestreift“. Mithin am schlimmsten sind ungleiche Zeilenabstände; der Leser weiß nicht, ob ein Absatz endet oder ob es sich um ein Zufallsprodukt handelt. Im Folgenden wird erläutert, wie $\text{T}_{\text{E}}\text{X}$ mit Zeilenabständen umgeht.

Im traditionellen Bleisatz ergab sich der typische Zeilenabstand aus der Höhe der einzelnen Lettern (*Kegelhöhe*). Um diesen Abstand (*Durchschuss*) zu erhöhen, wurde zwischen den einzelnen Zeilen Blindmaterial eingefügt. Eine Verringerung unter die Kegelhöhe war nur durch Abhobeln der Lettern mög-

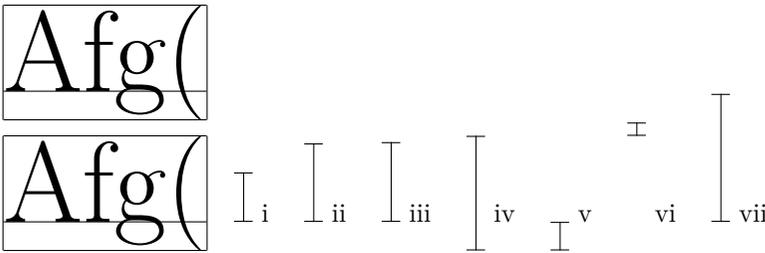


Abbildung 1: Die wichtigsten Begriffe im Zusammenhang mit Zeilenabständen. (i) x-Höhe (*ex*), (ii) Höhe einer Versalie (*Cap height*), (iii) Höhe einer Minuskel mit Oberlänge (*Ascender height*), (iv) Kegelgröße, (v) „Tiefe“ (*depth*), (vi) Durchschuss (*lineskip*), (vii) Zeilenabstand (*baselineskip*)

lich. Bei Schreibmaschine, Textverarbeitung und typischen DTP-Programmen wird der Zeilenabstand ebenfalls fest eingestellt.¹

Wie T_EX vorgeht

T_EX arbeitet anders. Es berücksichtigt statt der Kegelhöhe die tatsächliche Höhe der Zeile. Diese kann von der Kegelhöhe durchaus verschieden sein. Enthält eine Zeile etwa keine Buchstaben mit Unterlängen, ist sie für T_EX deutlich kleiner. Jeder Buchstabe hat einen Referenzpunkt. Dieser liegt im allgemeinen links unten auf der Grundlinie. Der Parameter `\baselineskip` gibt an, wie weit die Referenzpunkte voneinander entfernt sein sollen. Er sollte etwas größer als die Kegelhöhe der verwendeten Schrift eingestellt werden. Plain T_EX verwendet für die 10 Punkt Computer Modern einen Wert von 12 Punkt. Mittels `\lineskiplimit` wird angegeben, wann T_EX den Zeilenabstand erweitern soll. Wenn der Durchschuss unter diese Grenze fällt, erhöht T_EX den Durchschuss um das per `\lineskip` angegebene Maß. T_EX gewährleistet so, dass auch Zeichen außergewöhnlicher Größe nicht in benachbarte Zeilen hereinragen – siehe Abbildung 2C. Die Aufweitung der Zeilen, die dadurch erfolgt, ist allerdings in vielen Fällen unerwünscht.

¹ Die Kegelhöhe hat übrigens mit der Höhe der einzelnen Schriftzeichen nur wenig zu tun. In vielen Schriften ist die Oberlänge von Kleinbuchstaben etwas größer als die Höhe von Großbuchstaben, bei der Computer Modern übertreffen die Klammern die Buchstaben in der Höhe wie auch in der Tiefe.

	Sollen wir		Sollen wir		Sollen wir
	nennen		nennen		nennen
Sollen wir	was uns	Sollen wir	was uns	Sollen wir	was uns
nennen	zusagt?	nennen	zusagt?	nennen	zusagt?
was uns	Östhetik...	was uns	Östhetik...	was uns	Östhetik...
zusagt?		zusagt?		zusagt?	
Östhetik...		Östhetik...		Östhetik...	
(A)	(B)	(C)	(D)		

Abbildung 2: Verschiedene Einstellungen der Parameter für den Zeilenabstand: (A) gänzlich ohne Abstände, (B) mit Zeilenabstand 13 pt (C) mit 10,5 pt Abstand, (D) mit 10,5 pt Abstand und Korrektur des Abstandes der letzten beiden Zeilen. Die feinen Linien sind im jeweils idealen Zeilenabstand gezogen.

Probleme beim Textsatz

Wann tritt nun eine solche Aufweitung auf? In erster Linie dann, wenn Zeichen aus anderen Schriften untergemischt werden oder Mathematik gesetzt wird, leider jedoch auch, wenn Großbuchstaben mit Akzenten versehen werden. Diese Problematik ist schon sehr alt. In Dokumenten aus Bleisatz-Zeiten finden sich oft Beispiele, in denen Ä, Ö und Ü durch Ae, Oe und Ue ersetzt werden, um die Verwendung von Zeichen, die Großbuchstaben in der Höhe überragen, zu vermeiden. Während dort aber nur der Eindruck innerhalb der Zeile stört, verdirbt $\text{T}_{\text{E}}\text{X}$ durch das Aufweiten einzelner Zeilen das Bild der ganzen Seite. `german.sty` holt die Akzente etwas näher an die Buchstaben heran. Dadurch ist der Effekt nicht so gravierend wie in unverändertem $\text{T}_{\text{E}}\text{X}$. Dennoch empfiehlt sich für deutschen Satz generell eine leichte Aufweitung der Zeilenabstände, wie in Abbildung 2B. Da im Deutschen nur Kleinbuchstaben ohne Oberlänge und Großbuchstaben mit Akzenten versehen werden, kann die Erhöhung auch dadurch gemildert werden, dass eine Schrift, bei der die Großbuchstaben deutlich kleiner als Kleinbuchstaben mit Oberlänge sind, verwendet wird. Als angenehmer Nebeneffekt wird damit auch die Unruhe im Satzbild, die durch die im Deutschen übliche Großschreibung vieler Wörter entsteht, etwas gemildert.

Die Parameter für den Zeilenabstand werden von $\text{T}_{\text{E}}\text{X}$ erst bewertet, wenn der Absatz zuende ist. Werden diese Parameter also mehrmals innerhalb eines Absatzes verändert, gilt die zuletzt vorgenommene Einstellung.

Jenseits von gewöhnlichem Text

Werden Tabellen mit vertikalen Linien gesetzt, so ist es wichtig, dass die Linien nahtlos ineinander übergehen. Unterbrechungen zwischen einzelnen Zeilen sehen nicht gut aus. Hierzu wird der Abstand zwischen angrenzenden Zeilen auf 0 gesetzt, sinnvollerweise durch die Verwendung von `\offinterlineskip`. Um zu verhindern, dass Zeichen aneinander kleben oder dass Zeilenabstände unregelmäßig werden, wird dann in jede Zeile ein sehr großes Element gestellt. Hierfür wird `\strut` verwendet. \TeX setzt dann eine *box* der Breite 0 mit Höhe 8,5 Punkt und Tiefe 3,5 Punkt. Damit werden die 12 Punkt des üblichen Zeilenabstandes erreicht. Wird eine andere Schriftgröße und damit ein anderer Zeilenabstand verwendet, sollte `\strut` durch eine andere, von der Größe her angepasste Konstruktion ersetzt werden, beispielsweise durch eine `\vrule` der Breite 0. Auch außerhalb von Tabellen gibt es gute Einsatzmöglichkeiten für `\strut` oder ähnliche Konstruktionen, zum Beispiel wenn verschiedene *boxes* ‚von Hand‘ gestapelt werden.

Vorteile der Flexibilität

`\baselineskip` ist \TeX nisch gesehen keine `\dimen`, sondern `\glue`. Damit ist es möglich, \TeX etwas Spielraum bei den Zeilenabständen zu gewähren. Auf diesem Wege lässt sich erreichen, dass zum Beispiel die berühmte einzelne Zeile (Hurenkind) noch auf die vorige Seite gequetscht wird oder dass eine fast volle Seite gänzlich gefüllt wird. Bei beidseitig bedrucktem Papier möchte ich jedoch von solchen Vorhaben abraten, ebenso dann, wenn Doppelseiten betrachtet werden. Die unterschiedlichen Abstände fallen dann zu sehr auf. In solchen Fällen ist es vielleicht besser, einen Absatz um eine Zeile zu verlängern, indem `\looseness=1` angegeben wird.

Befehle zur Beeinflussung der Zeilenabstände

Plain- \TeX stellt eine Reihe von Befehlen und Parametern zur Verfügung, mit denen Zeilenabstände gezielt beeinflusst werden können. In \LaTeX spielt das Thema Abstände keine große Rolle, da \LaTeX vom Benutzer nicht erwartet, dass er an den Abständen „herumschraubt“.

`\baselineskip` ist der normale Abstand zwischen den Referenzpunkten aufeinanderfolgender Zeilen. Die folgende Anweisung setzt diesen Abstand auf 12 Punkt: `\setlength{\baselineskip}{10pt plus 1pt minus 0.5pt}`. Falls

$\text{T}_{\text{E}}\text{X}$ möchte, kann es den Abstand um bis zu einen Punkt dehnen und um bis zu 0,5 Punkt stauchen.

`\lineskip` Um diesen Betrag wird der Zeilenabstand erhöht, wenn der in `\lineskiplimit` angegebene Durchschuss unterschritten werden würde.

`\nointerlineskip` verhindert, dass nach der aktuellen Zeile ein Zeilenabstand eingefügt wird.

`\normalbaselines` setzt die Zeilenabstände auf Standardwerte; es macht sozusagen `\offinterlineskip` rückgängig.

`\offinterlineskip` setzt die Zeilenabstände generell auf 0 Punkt.

`\openup` erhöht die Zeilenabstände um das angegebene Maß. Ein typisches Beispiel: `\openup 1pt` erhöht den Durchschuss um einen Punkt.

`\parskip` gibt den vertikalen Abstand an, der nach einem Absatz zusätzlich gesetzt werden soll.

`\prevdepth` enthält die Tiefe der letzten gesetzten *box*.

`\strut` fügt eine unsichtbare Box der Höhe 8,5 Punkt und der Tiefe 3,5 Punkt ein.

`\vadjust` bringt das als Argument angegebene Material direkt unterhalb der aktuellen Zeile unter. Beispiel: `\vadjust{\vskip3pt}` fügt einmalig 3 Punkt zusätzlichen Durchschusses unter der Textstelle ein, an der es verwendet wird.

`\vskip` weist $\text{T}_{\text{E}}\text{X}$ an, vertikalen Abstand im angegebenen Umfang einzufügen. Diese Anweisung sollte nicht innerhalb von Absätzen verwendet werden; es handelt sich nämlich um ein vertikales Kommando; der Absatz wäre damit beendet.

Zeilenabstände in \LaTeX

Das Kommando `\selectfont` aus dem NFSS wählt nicht nur Schriftart und -größe, sondern auch den Zeilenabstand aus. Damit wird der häufigste Grund, Zeilenabstände zu verändern, behandelt. `\vspace` fügt zwischen Absätzen vertikalen Abstand ein. Für alle weiteren Zwecke stehen die oben beschriebenen Befehle und Parameter aus dem Fundus von Plain- $\text{T}_{\text{E}}\text{X}$ zur Verfügung.

Zukunftswünsche

Bei einer Schreibmaschine sind die Zeilenabstände in festen Stufen vorgegeben. $\text{T}_{\text{E}}\text{X}$ ist dieses Konzept fremd. In manchen Fällen wäre es aber vorteilhaft, wenn Zeilen an bestimmte ‚magnetische‘ Positionen auf der Seite rutschen könnten.

$\text{T}_{\text{E}}\text{X}$ berücksichtigt bei jeder Zeile nur die *maximale* Ausdehnung. Wenn an den Stellen, an denen ein hohes Element aus der unteren Zeile in den Fußraum der oberen Zeile hineinragt, in der oberen Zeile keine Unterlänge vorliegt, könnten die beiden Zeilen näher zusammengerückt werden, wie in Abbildung 2D gezeigt. Noch besser wäre, wenn im Falle einer solchen Kollision das Satzprogramm den Zeilenumbruch so ändern könnte, dass diese Buchstaben mit Unterlängen in der oberen Zeile und Zeichen mit großen Oberlängen in der unteren Zeile nicht mehr direkt untereinander stehen. Beides ist mit $\text{T}_{\text{E}}\text{X}$ nicht möglich, für den Nachfolger $\mathcal{N}\mathcal{T}\mathcal{S}$ jedoch in der Diskussion.

Der optimale Durchschuss richtet sich vor allem nach der Kegelgröße einer Schrift, aber auch nach anderen mit der Schrift verbundenen Eigenschaften. Eine fette Schrift braucht größere Abstände als eine magere, beispielsweise. Im TFM-Format, das $\text{T}_{\text{E}}\text{X}$ alle Informationen über die jeweilige Schrift zur Verfügung stellt, ist leider keine Angabe über den empfohlenen Zeilenabstand vorgesehen. Die Angabe `ex` kann diese Information nicht ersetzen, da es viele Schriften gibt, die bei gleicher Kegelhöhe unterschiedliche x-Höhen aufweisen (zum Beispiel Times und Arial). Auch hier wäre eine Nachbesserung wünschenswert.

Schlusswort

$\text{T}_{\text{E}}\text{X}$ ist hinsichtlich der Zeilenabstände wesentlich flexibler als traditionelle Systeme. Dies bringt leider nicht nur Vorteile mit sich; nicht zuletzt verpflichten die vielfältigen Parameter auch dazu, diese sinnvoll einzusetzen. Mit den für $\mathcal{N}\mathcal{T}\mathcal{S}$ diskutierten zusätzlichen Möglichkeiten wird die Qualität noch weiter gesteigert werden können.

Darstellung von Membranprotein-Topologien mit dem \TeX topo-Paket

Eric Beitz

Wie zahlreiche Reaktionen auf das \TeX shade-Paket für Nukleotid- und Proteinsequenzvergleiche (DTK4/99, [?]) zeigen, findet \TeX auch unter Biowissenschaftlern seine Anwender. Nun ist das nächste \TeX -Makropaket für diesen Nutzerkreis fertig gestellt. \TeX topo verarbeitet Daten zur Topologie von Membranproteinen und stellt sie schematisch dar. Dabei arbeitet es eng mit oben genanntem \TeX shade-Paket zusammen, um automatisierte Schattierungen zu ermöglichen. Beide Pakete zusammen genommen bilden den Ausgangspunkt des Bio \TeX -Projekts, das ausgereifte \TeX -Makros für Biochemiker/(Molekular-)Biologen zur Verfügung stellen will. Online-Dokumentationen und weitere Informationen finden sich unter der URL <http://homepages.uni-tuebingen.de/beitz>.

Hintergrund

Die Membran einer jeden Zelle ist gespickt mit Tausenden von Eiweißmolekülen, die verschiedenste Aufgaben und Gestalten haben. Zu nennen sind hier zum Beispiel Hormonrezeptoren, Kanäle, Transporter und Enzymkomplexe. Solche Proteine sind meist in der Membran verankert, indem sie diese mindestens einmal durchspannen (Proteine sind unverzweigte Ketten von Aminosäuren). Diese Membran-Domänen weisen eine charakteristische Spiralstruktur (α -Helix) und Aminosäurezusammensetzung (fettliebende = lipophile) auf. Weiterhin ist die Länge dieser Regionen auf einen Bereich von ≈ 21 Aminosäuren beschränkt, was auf die relativ konstante Dicke der Zellmembran von 7 nm zurück zu führen ist. Weitere Strukturmerkmale sind die Verbindungsstücke zwischen den Membrandurchgängen, so genannte Schleifen, und die Enden der Eiweißkette. Die Termini sind unterschieden durch chemischfunktionelle Gruppen. An einem Eiweißende – dem eigentlichen Proteinsynthesestart – befindet sich eine Aminogruppe (NH_2), der N-Terminus, am anderen Ende eine Carbonsäuregruppe (COOH), der C-Terminus. Die Topologie ist nun eindeutig definiert durch die Lokalisation des N-Terminus (intra- oder extrazellulär) und die Positionen der Membrandurchgänge.

Durch bloßes Aneinanderreihen von Aminosäuren ist die Proteinsynthese jedoch nicht abgeschlossen. Vielmehr unterliegen die meisten Eiweiße sogenann-

Alanin	Ala	A	Leucin	Leu	L
Arginin	Arg	R	Lysin	Lys	K
Asparagin	Asn	N	Methionin	Met	M
Asparaginsäure	Asp	D	Phenylalanin	Phe	F
Cystein	Cys	C	Prolin	Pro	P
Glutamin	Gln	Q	Serin	Ser	S
Glutaminsäure	Glu	E	Threonin	Thr	T
Glycin	Gly	G	Tryptophan	Trp	W
Histidin	His	H	Tyrosin	Tyr	Y
Isoleucin	Ile	I	Valin	Val	V

Tabelle 1: Der Drei- und Ein-Buchstaben-Code der 20 Aminosäuren.

ten Sekundärmodifikationen, wie zum Beispiel dem Anhängen von Lipiden zur weiteren Verankerung in der Membran, von Phosphatgruppen oder von verzweigten Zuckerketten.

Motivation

Heutige Protein-Datenbanken enthalten Informationen zur Topologie und zu Sekundärmodifikationen von nahezu 100.000 Proteinen (SwissProt-Datenbank, [?]). Außerdem sind recht zuverlässige Computerprogramme basierend auf Wissensdatenbanken (PROSITE, [?]) und der Bewertung von Aminosäureeigenschaften (PHD-Algorithmus, [?]) zur Vorhersage frei im Internet verfügbar. Diese Quellen haben ein standardisiertes ASCII-Ausgabeformat, was naturgemäß optimal computer-lesbar, aber ungenügend für das menschliche Auge ist.

Interessanterweise gibt es kaum Software zur automatisierten schematischen Darstellung dieser Daten. Daher werden die Zeichnungen mühsam mit Standard-Grafikprogrammen erstellt. Es hat sich eine Konvention etabliert, die 20 Aminosäuren mit ihrem Ein-Buchstaben-Code (siehe Tabelle 1) in einem Kreis zu symbolisieren und Sequenzbereiche von besonderem Interesse durch andere Symbole wie Rauten und Quadrate oder durch farbige Schattierungen hervorzuheben. Dies sind Aufgaben, die sich hervorragend in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ verwirklichen lassen. Es unterstützt die nötigen Befehle, um ASCII-Dateien einzulesen und zu analysieren. Mit der `picture`-Umgebung zur grafischen Ausgabe erhält man die gewohnt hohe $\text{T}_{\text{E}}\text{X}$ -Qualität. Durch Verwendung von `color.sty` werden aussagekräftige Schattierungen möglich.

Arbeitsweise von $\text{T}_{\text{E}}\text{X}$ topo und Systemvoraussetzungen

Das $\text{T}_{\text{E}}\text{X}$ topo-Paket stellt zwei neue $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Umgebungen bereit: (a) die `textopo`-Umgebung zum Plotten von Membranprotein-Topologien und (b) die `helicalwheel`-Umgebung zum Zeichnen von α -helikalen Membrandurchgängen als Aufsicht auf die Zellmembran oder vom Zellinneren aus gesehen. Innerhalb dieser Umgebungen sind über 60 neue Befehle anwendbar, um die Ausgabe wunschgemäß zu gestalten. Das Paket enthält eine mit Beispielen versehene, ausführliche Dokumentation aller Kommandos.

Als Quellen für Sequenz-, Topologie- und Sekundärmodifikationsdaten sind SwissProt-Datenbankeinträge, PHD-Dateien und Sequenz-Alignments verwendbar. Auch die manuelle Eingabe innerhalb der neuen Umgebungen ist möglich.

Zum Betrachten der Ausgabedatei ist ein PostScript-fähiger DVI-Viewer nötig, da die Verwendung von Anweisungen für Farbe und Rotation intern PostScript-Befehle erfordern. Alternativ kann die DVI-Datei auch komplett nach PostScript gewandelt werden, um sie dann mit einem PostScript-Viewer wie z. B. Ghostview zu betrachten oder zu drucken.

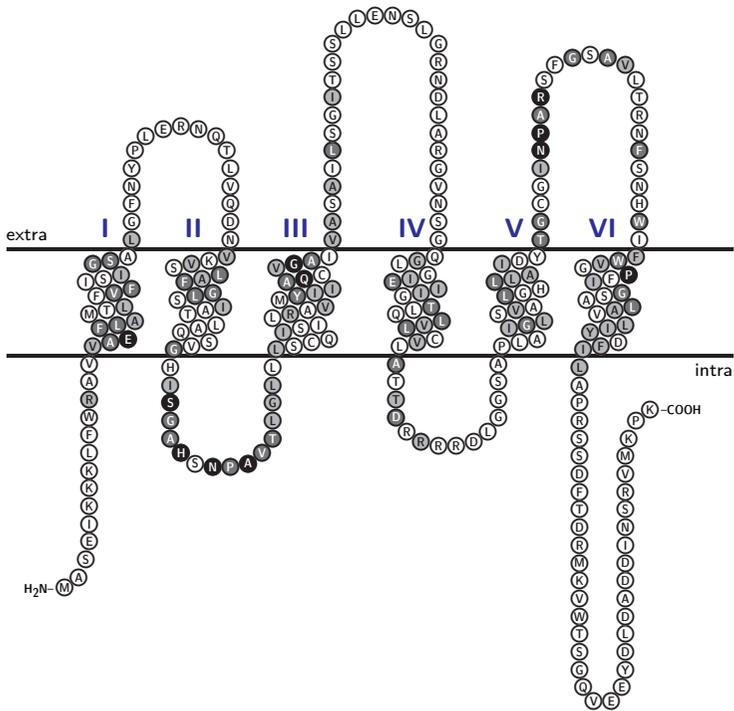
Ein erster einfacher Topologie-Plot

Für den Fall, dass die Topologiedaten in Form einer PHD-Datei vorliegen und ebenso ein Proteinsequenzvergleich zur Feststellung von konservierten Aminosäuren vorhanden ist, lässt sich mit wenigen Befehlen bereits ein recht ansehnlicher Plot zeichnen. Die folgende Umgebungsdefinition liefert die Ausgabe in Abbildung 1:

```
\begin{textopo}
  \getsequence{PHD}{AQP1.phd}
  \applyshading{similar}{AQP.MSF}
  \allmatchspecial
\end{textopo}
```

Die Zellmembran ist durch die beiden horizontalen Linien angedeutet. Das Protein durchspannt die Membran sechs mal, die beiden Termini liegen intrazellulär.

Mit minimalen Änderungen lassen sich die Membrandurchgänge als so genannte „helical wheels“ darstellen (Abbildung 2). In der Umgebung wurde



- ⊗ similar positions
- ⊗ conserved positions
- ⊗ invariable positions

Abbildung 1: Ein einfacher Topologie-Plot am Beispiel eines Wasserkanals (Aquaporin-1). Die Schattierung wurde automatisch mit dem $T_{E}X$ shade-Paket berechnet. Sie zeigt den Grad der Konservierung jeder Aminosäureposition innerhalb der Aquaporin-Familie an.

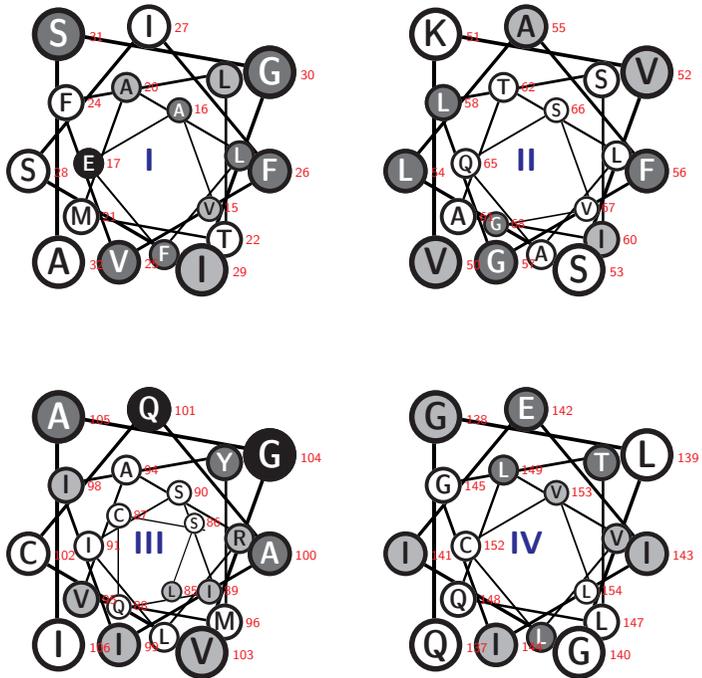


Abbildung 2: „Helical wheel“-Darstellung der ersten vier Membrandurchgänge von AQP1 mit Blick vom Extrazellulärraum auf die Membran.

zusätzlich der Befehl zur Unterdrückung der Legende gegeben und die Ausgabe auf die ersten vier Helices beschränkt:

```
\begin{helicalwheel}{1,2,3,4}
  \getsequence{PHD}{AQP1.phd}
  \applyshading{similar}{AQP.MSF}
  \allmatchspecial
  \hidelegend
\end{helicalwheel}
```

Einflussmöglichkeiten auf die Proteinstruktur

Verschiedene Faktoren können sich auf die Struktur eines Membranproteins auswirken, z. B. die schon genannten Lipidanker oder Sequenzabschnitte, die sich als „Halbschleifen“ in die Membran einlagern. Auch künstliche durch Klonierungstechniken und Rekombination eingeführte Modifikationen beeinflussen die Struktur, wie die Anheftung von zusätzlichen Bereichen an die Termini, sogenannte „tags“. Solche Anhängsel werden zur leichteren Aufreinigung eines Proteins oder zur Erkennung durch Antikörper eingeführt.

Alle beschriebenen Modifikationen lassen sich mit wenigen Befehlen in $\text{T}_{\text{E}}\text{X}$ -topo durchführen. Abbildung 3 zeigt einige Beispiele. Nachfolgend sind die Definitionen gezeigt, die den Darstellungen in Abbildung 3 zugrunde liegen, um die Mächtigkeit der neuen Befehle zu verdeutlichen.

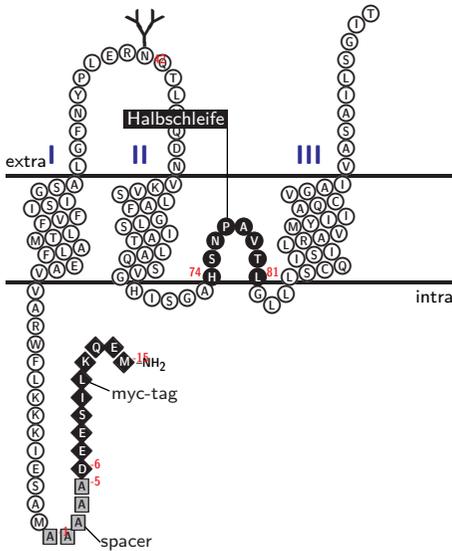
Quelltext zu Abbildung 3A:

```
\begin{textopo}
  \sequence{MASEIKKKLFWRAVVAEFLAMTLFVVFISIGSALGFNYPLERNQTL
  VQDNVKVSLAFGLSIATLAQSVGHISGAHSNPAVTLGLLLSCQISILRAVMYIIA
  QCVGAIVASAILSGIT}
  \MRs{15..32,50..68,74..81,85..106}
  \loopextent{15}
  \labelstyle{spacer}{box}{Black}{Gray30}{Black}{}
  \labelstyle{myc}{diamond}{Black}{Black}{White}{}
  \labelstyle{halfloop}{circ}{Black}{Black}{White}{}
  \addtagtoNterm[SE,8]{AAAAA}{spacer}{spacer}
  \addtagtoNterm[ESE,8]{MEQKLISEED}{myc}{myc-tag}
  \labelregion[N,30]{74..81}{halfloop}
                                {box[Black]:Halbschleife[White]}
  \glycosylation{42[ESE]}
  \flipNterm
  \hideCterm
  \hidelegend
\end{textopo}
```

Quelltext zu Abbildung 3B:

```
\begin{textopo}
  \sequence{FGSATAPFVL[LMLLASLNSCTNPWIYASF]SSSVSELRSLLCC}
```

A



B

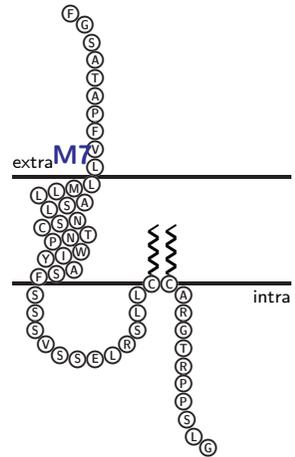


Abbildung 3: Beispiele für Strukturmodifikationen.

- A. Die vordere Hälfte von AQP1 ist gezeigt mit „tag“ am N-Terminus und der Poren-Halbschleife. Zusätzlich ist eine Zuckerkette angedeutet und der N-Terminus nach innen gefaltet.
- B. zeigt einen Ausschnitt des Vasopressin-Rezeptors (V_2) mit zwei Lipidankern.

```

ARGTRPPSLG}
\Nterm{extra}
\anchor{42,43}
\hideNterm \hideCterm
\loopextent{15}
\labelTM{1}{M7}
\end{textopo}

```

Markieren von Sequenzabschnitten

Neben der automatischen Schattierung lassen sich auch Bereiche „von Hand“ hervorheben, wie z. B. in Abbildung 3A an der Halbschleife gezeigt. Dabei ist zunächst die Form des Symbols wählbar. Zur Zeit stehen Kreis, Raute und Quadrat zur Verfügung. Danach kann für den Rand und die Füllung der Symbole sowie für die Lettern eine Farbe bestimmt werden. Mit Hilfe von `color.sty` lassen sich 64 vordefinierte PostScript-Farben verwenden. Einer eigenen Erweiterung steht jedoch nichts im Wege. `TEXtopo` ergänzt die Palette um Graustufen im 5%-Abstand. Schließlich kann ein beschreibender Text an den markierten Bereich geheftet werden, der wiederum farbig und unterlegt sein kann. Falls gewünscht, erscheint ein weiterer Text in der Legende.

Ausblick

Hier konnte nur ein grober Überblick über die Funktionen von `TEXtopo` gegeben werden. Es wird auf die Paket-Dokumentation verwiesen, die auf alle sonstigen Möglichkeiten der Feinjustierung von Schleifen, der Rotation von Helices, weiterer Beschriftungen und der Nutzung der `TEXshade`-Kompatibilität eingeht.

Durch die rege Nachfrage zu `TEXshade` ermutigt wurde dieses `TEXtopo`-Paket rasch fertig gestellt und die Idee zum `BioTEX`-Projekt geboren. Warum sollen nicht auch Biochemiker wohl formatierte `LATEX`-Dokumente vorweisen können? Beiträge zu `BioTEX` sind willkommen!

L^AT_EX für Vereinsmeier: *minutes.sty*

Knut Lickert

Mit dem hier beschriebenen Paket `minutes.sty` können Protokolle für Vereine o. ä. erstellt werden. Zusammen mit der Klasse `scrartcl` werden einzelne Protokolle erstellt, mit `scrreprt`-artigen Klassen kann eine Zusammenfassung mehrerer Protokolle erstellt werden.

Das Paket ist im CTAN unter `tex-archive/macros/latex/contrib/supported/minutes/` verfügbar.

Einleitung

Wenn man wie ich in einem Verein aktiv tätig ist, wird man wohl das Problem kennen: Man hat Vereinssitzungen und will Protokolle über die Sitzungen haben. Erst muss ein Protokollant gefunden werden, und dann stellt sich die Frage wie das Protokoll erstellt wird.

Dabei sammelt sich eine Vielzahl von einzelnen Protokollen in verschiedenen Dateiformaten (meist auf der Basis von winzigweich-Produkten, aber auch reine Handschriften) und unterschiedlichen Formatierungen an (mit oder ohne Inhaltsverzeichnis. . .). Diese Protokolle werden gesammelt und abgeheftet. Wenn man etwas sucht, müssen alle Protokolle durchgeblättert werden, um das Gesuchte zu finden.

Dieses Chaos führte in einem Verein, in dem ich Mitglied bin, dazu, dass keine Protokolle mehr erstellt wurden und wenn, diese nicht auffindbar waren, und wenn man sie doch gefunden hat, dort die gesuchte Information nicht gefunden wurde. Darauf habe ich in den sauren Apfel gebissen und beschlossen, ab sofort die Protokollierung selbst zu übernehmen.

Meine Anforderungen an die Protokolle waren:

- Jedes Protokoll hat einen identischen Kopf mit Titel, Sitzungsleiter, Protokollant, Anwesenden und Fehlenden. Sitzungsleiter und Fehlende sind optional.
- Alle Themen der Sitzung werden in einem Inhaltsverzeichnis aufgeführt.

- Die einzelnen Protokolle können ohne Probleme in eine Protokollsammlung übernommen werden. Diese Sammlung hat ein Gesamtverzeichnis aller Themen und einen Index.
- Protokolle anderer, in einem beliebigen Dateiformat, können eingefügt werden. Dazu wird ein Eintrag in das Inhaltsverzeichnis eingefügt und leere Seiten zur Aufnahme des fremden Protokolls freigelassen.
- Thesenpapiere und andere Einlagen können eingefügt werden, entsprechende Leerseiten werden freigelassen. Es wird ein Verzeichnis aller Zusätze erstellt.
- Aufgaben sollten Leuten zugeordnet werden und in Vorlagen die Leute solange quälen, bis sie erledigt werden.
- *Keine* Anforderung war, eine vorbereitete Tagesordnung mit einer realen Sitzung abzugleichen.

Diese Anforderungen bewogen mich, ein Paket `minutes.sty` zu schreiben. Eine erste Testversion hieß noch `protocol`. „Protocol“ existiert zwar auch im englischen, bezieht sich aber auf das diplomatische Protokoll, nicht auf Besprechungsprotokolle.

Alle Makros sind englisch definiert. Für deutschsprachige Protokollanten sind deutschsprachige Makros verfügbar. Dieser Artikel verwendet die deutschsprachigen Makros.

`minutes.sty` und Klassen

`minutes.sty` ist als Paket implementiert und greift intern auf Gliederungskommandos wie `\section` zurück. Durch die Wahl einer Klasse kann so weitgehend das Aussehen der Protokolle bestimmt werden. `minutes.sty` selbst hat zwei Möglichkeiten, die das Aussehen beeinflussen. Mit den Optionen `ListTitle` und `TableTitle` kann das Aussehen der Protokollköpfe beeinflusst werden. Mit der Option `TwoColumn` wird der Textkörper zweispaltig gesetzt, Überschriften werden über beide Spalten gedruckt.

Das Paket `minutes.sty` unterscheidet zwischen Einzelprotokollen und Protokollsammlungen. Wird das Paket zusammen mit `scrartcl` oder einer ähnlichen Klasse verwendet, wird ein einzelnes Protokoll ausgegeben. Zusammen mit `scrreprt` ist eine Sammlung mehrerer Protokolle möglich. Unterscheidungskriterium ist die Existenz des Makros `\chapter`. Will man den op-

```

\begin{Protokoll}{Beispiel eines Protokoll in ...}
\untertitel{Dieses Protokoll ist mit \minutes\ erzeugt}
\moderation{Knut Lickert}
\protokollant{Knut Lickert}
\teilnehmer{Leser von "'\DTK"'}
\sitzungsdatum{\today}
\sitzungsbeginn{20:00}
\sitzungsende{23:00}
\verteiler{alle Interessierten}
\fehlend[alle Analphabeten]{Vereinsm{\\"u}ller}
\protokollkopf
\end{Protokoll}

```

Abbildung 1: Ein Protokoll mit *minutes.sty*

tischen Eindruck einer neuen Klasse mit den verschiedenen Optionen von *minutes.sty* testen, liegt dem Paket eine Beispieldatei `MinutesStyleGuide` bei, die einem eine schnelle Übersicht über alle Optionen bietet.

Eine Ausgabe mehrerer Protokolle mit `scrartcl` ist prinzipiell möglich, das Ergebnis wird aber kaum den Wünschen entsprechen, denn jedes Protokoll erhält ein vollständiges Inhaltsverzeichnis aller Protokolle.

Die Protokoll-Umgebung

Ein Protokoll wird mit der Umgebung `Protokoll` definiert. Am Beginn der Umgebung werden Protokollant, Teilnehmer, ... definiert. Ein Beispiel ist in Abbildung 1 zu sehen. Per Option kann zwischen einem Protokollkopf in Listen- (mit der Option `ListTitle`, siehe Abbildung 2) oder Tabellenform (mit der Option `TableTitle`, siehe Abbildung 3) gewählt werden.

Tagesordnungspunkte

Ein Tagesordnungspunkt wird mit `\topic{Titel}` eingeleitet. Untergliederungen werden mit `\subtopic` und `\subsubtopic` definiert. Als Überschrift ohne Hierarchiestufe kann `\minitopic` verwendet werden.

So definierte Tagesordnungspunkte werden dem Protokoll als Tagesordnung vorangestellt. `\topic`-Einträge werden zusätzlich in das Inhaltsverzeichnis der Protokollsammlung übernommen.

1 Beispiel eines Protokollkopfs in Listform

Dieses Protokoll ist mit `Minutes.sty` erzeugt

Moderation Knut Lickert

Protokollant Knut Lickert

Anwesend Leser von „Die TeXnische Komödie“

Fehlend(entschuldigt) alle Analphabeten

Fehlend(unentschuldigt) Vereinsmüller

Datum 24. Juli 2000 20:00 – 23:00

Verteiler alle Interessierten

Tagesordnung

1.1	Top eins	2
1.1.1	zu Top eins	2
(1.1)	Einschub in Tagesordnung	2
1.2	Top zwei	2

Abbildung 2: Ein Protokollkopf in Listform

2 Beispiel eines Protokollkopfs in Tabellenform

Dieses Protokoll ist mit <code>Minutes.sty</code> erzeugt	
Datum: 24. Juli 2000 20:00– 23:00	
Moderation: Knut Lickert	Protokollant: Knut Lickert
Anwesend: Leser von „Die TeXnische Komödie“	Verteiler: alle Interessierten
Fehlend(entschuldigt): alle Analphabeten	Fehlend(unentschuldigt): Vereinsmüller

Tagesordnung

2.1	Top eins	4
2.1.1	zu Top eins	4
(2.1)	Einschub in Tagesordnung	4
2.2	Top zwei	4

Abbildung 3: Ein Protokollkopf in Tabellenform

Wird eine Einladung zu einer Sitzung versendet, so soll die versendete Tagesordnung identisch sein mit der Tagesordnung aus dem Protokoll. Trotzdem kommt es vor, dass Zusatzpunkte auftauchen. `minutes.sty` unterstützt solche Wünsche mit dem Makro `\zusatztopic`. Diese Einträge erhalten eine eigenständige eingeklammerte römische Nummerierung.

Geheimes

Sollen Teile eines Protokolls nicht veröffentlicht werden, kann der entsprechende Bereich zwischen mit `\begin{Secret}` und `\end{Secret}` geschachtelt werden. Diese geheimen Teile werden nur übernommen, wenn die Option `Secret` gewählt wurde.

Kurze Abschnitte können mit `\secret` verborgen werden.

Diskussionspapiere

Gibt es zusätzliche Papiere, die einem Protokoll hinzugefügt werden sollen (Thesenpapiere, Verträge etc.) kann man das Makro `\attachment` verwenden. Das Makro hat drei Parameter:

- eine optionale Kurzbezeichnung, die zum Setzen eines Labels benutzt wird,
- ein Titel, der im Inhaltsverzeichnis erscheint,
- die Seitenzahl des Anhangs.

```
\anhang[verweis]{Thesepapier mit zwei Seiten}{2}
\anhang{Thesepapier mit drei Seiten}{3}
```

ergibt im laufenden Text:

Anhang 0.i: Thesepapier mit zwei Seiten Seite 49[2]

Anhang 0.ii: Thesepapier mit drei Seiten [3]

Die Anhänge werden römisch nummeriert. Mit `\ref{verweis}` kann in diesem Beispiel auf das erste Thesepapier verwiesen werden. („Anhang 0.i auf Seite 49“).

Mit `\anhangsliste` wird ein Verzeichnis mit allen Anhängen erstellt, siehe dazu Abbildung 4. Die Anhänge müssen manuell eingelegt und nummeriert

0.i	Thesepapier mit zwei Seiten	49 [2]
0.ii	Thesepapier mit drei Seiten	51 [3]

Nach Ausgabe der Anhangsliste wird ein Seitenumbruch durchgeführt. In diesem Beispiel erscheint die Liste auf Seite 48, die Anhänge beginnen auf der nächsten Seite. Die Seitennummerierung nach dieser Seite ginge somit mit 54 weiter. Für „Die T_EXnische Komödie“ wird das unterdrückt.

Abbildung 4: Anhangsliste

werden. In der Anhangsliste wird die Seitennummer und die Anzahl der Seiten angegeben. Das Verzeichnis erhöht die weitere Seitennummerierung gemäß der Seitenanzahl der Anhänge.

Die Möglichkeit eines Anhangsverzeichnisses pro Protokoll ist für eine spätere Version geplant. Die Ausgabe ohne eine Änderung der Seitennummerierung soll per Option an- und ausschaltbar sein.

Aufgaben und Listen offener Aufgaben

Mit dem Makro `\aufgabe` werden Aufgaben definiert. Das Makro hat vier Parameter:

1. Erledigt-Datum (optional).

Ist dieser Parameter angegeben, wird das Datum in einer Fußnote vermerkt und die Aufgabe wird nicht mehr in der Liste der offenen Aufgaben aufgeführt. Erledigte Aufgaben müssen immer am Ort ihrer Definition als bearbeitet gesetzt werden.

2. Verantwortlicher. Ist niemand spezielles verantwortlich, kann `*` verwendet werden.
3. Datum, zu dem es erledigt werden muss (optional).
4. Was erledigt werden muss.

```
\aufgabe[7. Juni]{Knut}[vor Redaktionsschluss]%
      {Artikel schreiben}
\aufgabe*{Korrektur lesen}
\aufgabe{Redaktion}[vor Drucktermin]{\DTK\ fertigstellen}
\aufgabe[wird gerade erledigt]*{\DTK\ lesen}
```

Liste offener Aufgaben

Irgendwer	Korrektur lesen	0/50
Redaktion	vor Drucktermin	
	Die T _E Xnische Komödie fertigstellen	0/50

Abbildung 5: Liste der offenen Aufgaben

ergibt:

Aufgabe (vor Redaktionsschluss)¹: Artikel schreiben (Knut)

Aufgabe: Korrektur lesen

Aufgabe (vor Drucktermin): Die T_EXnische Komödie fertigstellen (Redaktion)

Aufgabe²: Die T_EXnische Komödie lesen

Das Makro `\aufgabenliste` erzeugt eine Liste aller noch offenen Aufgaben (siehe Abbildung 5). Als optionales Argument kann der Dateiname einer anderen Protokollsammlung angegeben werden. So kann man beispielsweise offene Aufgaben aus dem Vorjahr (`\aufgabenliste[Protokolle1999]`) übernehmen. Die so ausgegebenen Seitenzahlen beziehen sich auf die Seiten aus der Originaldatei.

Termine und Vereinskalendar

Termine können mit dem Makro `\termin{2000/12/24}{Heiligabend}` definiert werden. Die normierte Eingabe mit Jahr, Monat und Tag, jeweils getrennt durch „/“, wird in eine sprachabhängige Ausgabe umgesetzt. Obiges Beispiel ergibt:

24. Dezember 2000: Heiligabend

Aus den Terminen kann direkt eine Eingabedatei für das Paket `calendar` erzeugt werden. Dieses Paket ermöglicht die Erstellung von Kalendern in verschiedenen Versionen. So ist es unter anderem möglich, monatliche Kalendarblätter oder eine Terminliste zu erzeugen. Das Paket `calendar` ist im CTAN unter `macros/latex/contrib/supported/calendar` zu finden.

¹ 7. Juni 2000

² wird gerade erledigt

Eine Eingabe von Umlauten (egal ob als „ä“ oder `\"a`) ist leider in der aktuellen Fassung nicht möglich. Soll keine Eintragung in einen Kalender stattfinden, ist das Makro `\termin*` zu verwenden.

Ausblick und zukünftige Versionen

Es liegt schon ein Makro `\beschluss` für Beschlüsse vor. Eine rudimentäre Beschlussliste ist schon machbar, muss aber noch von mir überarbeitet werden. Mein Wunsch ist eine Gruppierung von Beschlüssen nach Themen. Vermutlich ist dies nur mit einer Nachbearbeitung mit `makeindex` oder ähnlichem möglich.

Aufgaben sollen analog wie Termine direkt in einen Kalender eingefügt werden können. Das Makro `\aufgabe` soll so erhalten bleiben, wie es jetzt ist; genau datierbare Aufgaben sollen einem neuen Makro vorbehalten bleiben.

Bekannte Fehler sollen selbstverständlich beseitigt werden. Dies sind momentan die Unmöglichkeit von Umlauten in Terminen in Kombination mit Kalendern und Instabilität der `Secret`-Umgebung zusammen mit der mehrspaltigen Ausgabe (Option `TwoColumn`).

Ich würde mich freuen, wenn das Paket auch von anderen „Vereinsmeiern“ genutzt wird. Über Rückmeldungen, wo das Paket genutzt wird, freue ich mich besonders. Kommentare zu Sinn und Unsinn der Makros, sowie weitere Verbesserungsvorschläge nehme ich gerne entgegen.

Umbruch in schmalen Spalten

Gerd Neugebauer

Beim Tabellensatz kann es vorkommen, dass längerer Text in schmalen Spalten gesetzt werden soll. Das ist der Fall, wenn eine Tabellenzelle nicht nur einen einfachen Wert oder ein Wort enthält, sondern ein gesamter Absatz darin enthalten ist, der umbrochen wird. Da Tabellenspalten in der Regel sehr schmal sind, wird es \TeX Probleme bereiten, solch einen Absatz zu setzen.

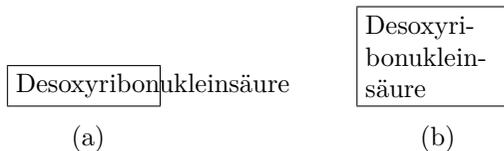


Abbildung 1: Umbruch des ersten Wortes eines Absatzes

Normalerweise versucht $\text{T}_{\text{E}}\text{X}$ den Zeilenumbruch so zu machen, dass eine möglichst „gute“ Lösung für den gesamten Absatz gefunden wird. Dazu werden sowohl Trennstellen zwischen den Worten, als auch Trennungen innerhalb von Worten in Betracht gezogen – sofern die Schrift dies gestattet und Trennmuster aktiviert sind.

Dabei tritt ein merkwürdiger Effekt aus den Tiefen von $\text{T}_{\text{E}}\text{X}$ zutage. $\text{T}_{\text{E}}\text{X}$ sucht nämlich erst ab dem ersten „glue“ innerhalb eines Absatzes nach Trennstellen innerhalb von Wörtern. „glue“ wird von $\text{T}_{\text{E}}\text{X}$ an den Stellen automatisch eingesetzt, an denen etwas gedehnt oder gestaucht werden kann. Dieser „glue“ findet sich in einem Absatz normalerweise frühestens am Ende des ersten Wortes, da vor dem ersten Wort kein „glue“ eingefügt wird. Damit ergibt sich, dass das erste Wort in einem Absatz nicht getrennt wird. Ist dieses Wort sehr lang, wie „Desoxyribonukleinsäure“, so wird dieses Wort zu einer „overfull $\backslash\text{hbox}$ “-Warnung führen.

Dieses Verhalten ist auch in einem „normalen“ Absatz eines Buches durchaus sinnvoll, da damit die Untersuchung von Trennstellen im ersten Wort eines Absatzes eingespart werden kann. Aber bei schmalen Spalten wie in einer Tabelle, in Randbemerkungen ($\backslash\text{marginpar}$) oder beim mehrspaltigen Satz kann dies schon einmal ärgerlich sein.

Um dieses Problem zu umgehen, reicht es aus, am Anfang der Zeile „glue“ einzufügen, der an den Abständen nichts ändert. Das kann mit $\backslash\text{hspace}\{0\text{pt}\}$ geschehen. Der Erfolg kann in der Abbildung 1 im Beispiel (b) bewundert werden, während das Beispiel (a) das Originalverhalten zeigt.

Für das Makro $\backslash\text{marginpar}$ kann man so etwas sogar in ein eigenes Makro verpacken und somit automatisieren. Das kann dann folgendermaßen aussehen:

```
 $\backslash\text{let}\backslash\text{MarginPar}\backslash\text{marginpar}$   
 $\backslash\text{renewcommand}\backslash\text{marginpar}[1]\{\backslash\text{MarginPar}\{\backslash\text{hspace}\{0\text{pt}\}\#1\}$ 
```

Mit der ersten Anweisung wird die aktuelle Definition von `\marginpar` unter dem Namen `\MarginPar` gespeichert. Mit der zweiten Anweisung wird `\marginpar` dann neu definiert, wobei die alte Definition zum Einsatz kommt und vor das Argument der benötigte „glue“ eingefügt wird.

Natürlich erscheinen beim Setzen des obigen Beispiels (b) immer noch TeX-Fehlermeldungen, nämlich zwei „underfull `\hbox`“-Warnungen. Das kommt daher, dass TeX versucht, einen Blocksatz zu erreichen, aber das nun getrennte Wort nicht genügend auseinanderziehen kann. Es ist an solchen Stellen oftmals problematisch, auf Blocksatz zu bestehen. Ein Flattersatz oder Rauhsatz ist hier meistens adäquater.

TeX-Rätsel: Die Lösung

Dies ist die Lösung des TeX-Rätsels aus „Die TeXnische Komödie“ 2/2000.



Rezensionen

„Digital Typography“ von Donald E. Knuth

Torsten Reincke

Dieses Buch von Donald Knuth ist eine reichhaltige Textsammlung von Artikeln und Reden, die Knuth im Laufe der Geschichte von T_EX und METAFONT verfasst hat. Nur drei Kapitel sind neu, die anderen sind Nachdrucke aus verschiedensten Zeitschriften. In Knuths unvergleichlicher Art erfährt der Leser viel Interessantes, aber auch Kurioses zur digitalen Typographie seit 1977.

Eine neues Buch von Knuth? Fast. Jedenfalls war ich erfreut, als ich im letzten Jahr in der Uni-Buchhandlung von Yale in New Haven, Connecticut, stöberte und mir dieser Titel in der Computer- und Design-Abteilung in die Hände fiel. Ein kurzes Blättern, und mir war klar, dass ich es einfach kaufen musste.

Im Grunde ist jedoch fast keines der 34 (englischen) Kapitel wirklich neu. Bis auf das Vorwort, zwei fast autobiographische Abschnitte aus den ersten Tagen von T_EX (`texdr.aft` und `tex.one`) und das letzte Kapitel („The Final Errors of T_EX“) wurden alle früher schon einmal veröffentlicht. Wer aber nicht gerade die Ausgabe 19 (1985) von „Visible Language“ oder Heft 6 (1987) der „ACM Transactions on Graphics“ gelesen hat, kennt diese Artikel wohl kaum. Und auch nicht jeder hat alle TUGboat-Ausgaben gelesen, wo viele der hier zusammengefassten Aufsätze erschienen sind. Es findet sich sogar ein Bild von Knuth im Alter von gerade mal vier Jahren, lesend natürlich.

Dabei geht es bis ins Detail, wie zum Beispiel im Kapitel „Breaking Paragraphs Into Lines“, das allein 90 Seiten umfasst, oder in „The Letter S“. Wer jemals versucht hat, eine Schrift zu entwerfen, wird die Tipps und Erfahrungen dort zu schätzen wissen. Das S ist wirklich schwierig! Ein anderes Kapitel befasst sich ausführlich mit der Möglichkeit, spezielle Fonts für digitale Halbtonbilder zu benutzen. Dabei werden verschiedene Optimierungen

behandelt: Dithering, Floyd-Steinberg-Adaption, Dot-Diffusion oder Smooth-Error-Diffusion.

Ein richtiger Leckerbissen ist das Kapitel „AMS Euler – A New Typeface for Mathematics“. Knuth schrieb es zusammen mit Hermann Zapf, dem unermüdlichen Schriftkünstler. Es ging damals darum, eine Schriftfamilie zu schaffen, die nicht nur für mathematische Texte geeignet ist, sondern für die gesamte Wissenschaft. Es begann eine Zusammenarbeit, die es vielleicht so noch nicht gegeben hatte. Ein Typograph und ein Mathematiker setzten sich zusammen und entwickelten per E-Mail, Brief und Zusammentreffen die Schriftfamilie AMS Euler. Es ist interessant zu sehen, wie die Erwartungen der mathematischen Seite (die nicht nur aus Knuth bestand; es gab ein richtiges Komitee) und die ästhetischen Vorstellungen Zapfs zu einer fruchtbaren Entwicklungsarbeit führten. Dazu ist das Kapitel mit Abbildungen angereichert, die fast kalligraphisch sind. Zitat: „Unanimous approval: Bravo, bravo!“

In einigen Kapiteln bekommt der Leser Einsicht in die Entstehung und den Hintergrund von $\text{T}_{\text{E}}\text{X}$ und METAFONT. Der genannte Abschnitt `texdr.aft` ist zum Beispiel ein „original source document“. Es beginnt mit einem Tagebucheintrag vom 7. Februar 1977 und schildert von da an Bruchstücke der Entstehungsgeschichte von $\text{T}_{\text{E}}\text{X}$. Besonders interessant ist die von Knuth damals als Memo an sich selbst geschriebene vorläufige Beschreibung von $\text{T}_{\text{E}}\text{X}$, eben die eigentliche Datei `texdr.aft`. Dort wird bereits recht detailliert beschrieben, wie $\text{T}_{\text{E}}\text{X}$ funktionieren soll. Viele Befehle waren als „keywords“ ohne den Rückstrich entworfen, andere, wie das Dollarzeichen für Mathematiksatz, haben bis heute noch die gleiche Form.

Auch Kurzweiliges ist dabei. Besonders gut hat mir das Kapitel „ $\text{T}_{\text{E}}\text{X}$ Incunabula“ gefallen. Es behandelt die Wiegendrucke der $\text{T}_{\text{E}}\text{X}$ -Ära. Die teilweise abgebildete Weihnachtsgeschichte ist wirklich wunderschön illustriert. Sie erschien zu Weihnachten 1978 und hatte eine Auflage von etwa 100 Exemplaren. Drei Kapitel „Questions and Answers“ sind dabei. Sie sind aus Treffen entstanden, die Knuth bei verschiedenen Konferenzen mit $\text{T}_{\text{E}}\text{X}$ -Benutzern aus aller Welt hatte. Knuths Antworten auf Fragen aller Art sind ausführlich und teilweise übermittelt er eine Art trockenen Witz, dass man wünscht, dabei gewesen zu sein.

Viele dieser Aufsätze haben trotz ihres teilweise wissenschaftlichen Hintergrundes einen verschmitzten Unterton oder enthalten immer wieder aufmunternde Bemerkungen. Die Materie wird nie langweilig und Knuth versteht es hervorragend *für* die Leser zu schreiben und nicht für das Papier, auf dem

sie gedruckt ist. Man fühlt sich oft wie in einem Zwiegespräch und nicht wie ein Belehrter.

Danke Don, für so viele Stunden vergnüglichen Lesens und Lernens!

Donald E. Knuth

Digital Typography

CSLI Lecture Notes Number 78

CSLI Publications Stanford, California, 1999

685 Seiten, mit vielen Abbildungen

ISBN 1-57586-010-4 (Gebunden), Preis 24,95 £

ISBN 1-57586-011-2 (Paperback), Preis 35 £

ADDITIVE-Anzeige

Spielplan

Termine

- 11.9.–12.9.2000** PODDP'00 – 5th International Workshop on Principles of Digital Document Processing
TU München, München
Der Workshop findet im Zusammenhang mit der DDEP 2000 direkt vor dieser statt.
- 13.9.–15.9.2000** DDEP 2000 – Digital Documents and Electronic Publishing 2000
TU München, München
Kontakt: Anne Brüggemann-Klein
- 6.10.–7.10.2000** Herbsttagung und 23. Mitgliederversammlung von DANTE e.V.
FernUniversität in Hagen
Kontakt: Thomas Feuerstack
- 7.3.–10.3.2001** DANTE 2001 und 24. Mitgliederversammlung von DANTE e.V.
Fachhochschule Rosenheim
Kontakt: Prof. Dr. Georg Lachenmayr
- 12.8.–15.8.2001** T_EX2001 – A T_EX Odyssey
University of Newark/Delaware
- 23.9.–27.9.2001** EuroT_EX
Kerkrade/Niederlande

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/dante/Stammtische.html>.

Berlin

Rolf Niepraschk
Tel.: 0 30/3 48 13 16
niepraschk@ptb.de
Gaststätte „Bärenschenke“
Friedrichstr. 124
Zweiter Donnerstag im Monat, 19.00 Uhr

Bremen

Martin Schröder
Tel.: 04 21/2 23 94 25
martin@oneiros.de
Universität Bremen, Hoyers Unikum
Erster Donnerstag im Monat, 18.30 Uhr

Chemnitz

Hans Friedrich Steffani
hans.steffani@e-technik.tu-chemnitz.de
Bahnhofstr. 8, 1. OG, Raum G104
Zweiter Mittwoch im Monat, 18.00 Uhr

Dortmund

Stephan Lehmke
Stephan.Lehmke@cs.uni-dortmund.de
Cafe Durchblick
Universität Dortmund, Campus Nord
Zweiter Mittwoch im Monat, 20.00 Uhr

Dresden

Hilmar Preuße
hille42@gmx.de
Cafe B'liebig
Liebigstr. 24
Letzter Mittwoch im Monat, 19.00 Uhr

Erlangen

Walter Schmidt, Peter Seitz
walter.schmidt@arcormail.de,
p.seitz@koehler-seitz.de
Gaststätte „Gambrinus“
Vierzigmannstr. 7
Dritter Dienstag im Monat, 20.00 Uhr

Freiburg

Heiko Oberdiek
Tel.: 07 61/4 34 05

oberdiek@ruf.uni-freiburg.de
Gaststätte „Aquila“
Sautierstr. 19
Dritter Donnerstag im Monat, 19.30 Uhr

Hamburg

Volker Hüttenrauch
volker_huettenrauch@hh.maus.de
Vereinsheim der Hamburger Microcomputer-Hochschulgruppe
Gründelallee 143 (Hinterhof)
Letzter Donnerstag im Monat, 18.00 Uhr

Hannover

Mark Heisterkamp
Regionales Rechenzentrum
Tel.: 05 11/7 62 51 34
heisterkamp@rrzn.uni-hannover.de
Seminarraum RRZN
Schloßwender Str. 5
Zweiter Mittwoch von geraden Monaten,
18.30 Uhr

Heidelberg

Luzia Dietsche
Tel.: 0 62 21/5 4 45 27
luzia.dietsche@urz.uni-heidelberg.de
China-Restaurant „Palast“, Lessingstr. 36
Letzter Mittwoch im Monat, 20.00 Uhr

Karlsruhe

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

München

Michael Niedermair
m.g.n@gmx.de
Gastwirtschaft „Rhaetenhaus“
Luisenstr. 27
Erster Dienstag im Monat, 19.00 Uhr

Münster

Johannes Reese
reesej@uni-muenster.de
Im Blauen Haus
Kreuzstr. 16/17
Erster Montag im Monat, 20.00 Uhr

Oldenburg
Konrad Blum
konrad@blum.physik.uni-oldenburg.de
Bistro „DaCapo“
Friedhofsweg 64
Zweiter Montag im Monat, 20.00 Uhr

Stuttgart
Marcus Schweizer
Tel.: 07 11/6 85 44 44
schweiz@theochem.uni-stuttgart.de
Gaststätte „Alte Mira“

Büchsenstr. 24
Zweiter Dienstag im Monat, 19.30 Uhr

Wiesbaden
Christian Kayssner
Tel.: 06 11/4 81 17
„Andreas Klause“
Elsässer Platz 3
Erster Montag im Monat, 20.00 Uhr

Wuppertal
Andreas Schrell
Tel.: 02 02/50 63 81
schrell@wupperonline.de
Croatia „Haus Johannisberg“
Südstr. 10, an der Schwimmpfer
Wuppertal-Elberfeld
Zweiter Donnerstag im Monat, 19.30 Uhr

Adressen

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

Tel.: 0 62 21/2 97 66 (Mo, Mi–Fr, 10⁰⁰–12⁰⁰ 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
Postbank Karlsruhe Auslandsüberweisungen
BLZ 660 100 75, Kontonummer 213 400 757

Beiträge:	ermäßigte Mitgliedschaft	60,- DM
	Privatmitgliedschaft	80,- DM
	Institutionen des öffentlichen Rechts und Forschungseinrichtungen	120,- DM
	Firmen, die T _E X anwenden	300,- DM
	Firmen, die Produkte in Verbindung mit T _E X anbieten	500,- DM

Präsidium

Präsident:	Thomas Koch	president@dante.de
Vizepräsident:	Volker RW Schaa	vice-president@dante.de
Schatzmeister:	Horst Szillat	treasurer@dante.de
Schriftführer:	Günter Partosch	secretary@dante.de
Beisitzer:	Klaus Höppner	adviser@dante.de

Server

ftp: [ftp.dante.de](ftp://ftp.dante.de) [134.93.8.251]
E-Mail: ftpmail@dante.de
WWW: <http://www.dante.de/>

Die T_EXnische Komödie

12. Jahrgang Heft 3/2000 September 2000

Impressum

Editorial

Hinter der Bühne

4 Grußwort

Bretter, die die Welt bedeuten

6 Schöner Lesen – gut lesbare und optisch ansprechende Texte durch gute Typografie

30 Zeilenabstände

36 Darstellung von Membranprotein-Topologien mit dem T_EXtopo-Paket

44 L^AT_EX für Vereinsmeier: `minutes.sty`

51 Umbruch in schmalen Spalten

53 T_EX-Rätsel: Die Lösung

Rezensionen

54 „Digital Typography“ von Donald E. Knuth

Spielplan

58 Termine

59 Stammtische

Adressen