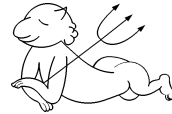


DANTE

Deutschsprachige Anwendervereinigung T_EX e.V.



Die



T_EXnische
Komödie

Ein Bühnenstück in (hoffentlich) vielen Folgen

Heft 1(1990)

2. Jahrgang

März 1990

Impressum

„Die T_EXnische Komödie“ ist die Mitgliedszeitschrift von DANTE, Deutschsprachige Anwendervereinigung T_EX e.V. Namentlich gekennzeichnete Beiträge geben die jeweilige AutorInnen-Meinung wieder.

Soweit nicht explizit anderweitig gekennzeichnet, ist die Reproduktion oder Nutzung der erschienenen Beiträge durch konventionelle, elektronische oder beliebige andere Verfahren nur im nicht-kommerziellen Rahmen gestattet. Verwendungen in größerem Umfang bitte zur Information bei DANTE melden.

Wenn Sie die in den Beiträgen beschriebene Software beziehen möchten, schicken Sie bitte an die unten angegebene Adresse einen adressierten und ausreichend frankierten Rückumschlag mit einer entsprechenden Anzahl formatierter Disketten.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 400

Herausgeber: DANTE, Deutschsprachige Anwendervereinigung T_EX e.V.
c/o Rechenzentrum der Universität
Im Neuenheimer Feld 293
D-6900 Heidelberg
e-mail: dante@dhdur1

Druck: Behr Fotodruck GmbH
Hasenbergstr. 82A
D-7000 Stuttgart 1

Redaktionsteam: Gabriele Kruljac (verantwortlich) und Barbara Burr

Redaktionsschluß für Heft 2/90: 15.5.90

Wir bitten, alle Beiträge an die DANTE-Anschrift zu senden.

Editorial

Nach der Generalprobe liegt hiermit nun also die Premiere der „T_EXnischen Komödie“ vor.

Nach dem, was so zu hören war, gefiel die Form der „T_EXnischen Komödie“ ganz gut; wir haben uns gedacht, erstmal bei dem Format zu bleiben, denn solange die Hefte nicht dicker sind, sind sie mit DIN A5 wenigstens „standfester“ als in DIN A4.

Nur der Inhalt: hier möchten wir uns bei allen entschuldigen, die wir mit dem im letzten Heft veröffentlichten Spielplan verwirrt haben. Wir hoffen, daß in dieser Neuauflage endgültig alle Fehler behoben sind.

Kurz vor knapp erreichten uns doch noch einige Beiträge; außerdem wollten wir alle, die nicht an der Mitgliederversammlung teilnehmen konnten in den Genuß eines möglichst aktuellen Protokolls bringen. Daher die kleine Verzögerung in der Fertigstellung und Auslieferung der Zeitschrift. Die Beiträge unserer Mitglieder in diesem Heft ermuntern hoffentlich auch andere, uns etwas zur Veröffentlichung zukommen zu lassen.

Vielleicht wird Ihnen diese Ausgabe etwas „englisch“ vorkommen. Aber wir hielten es für besser, sowohl die Tagungsankündigungen wie auch den Artikel von D. E. Knuth so original wie möglich zu belassen.

Gabriele Kruljac und Barbara Burr

Hinter der Bühne

(Vereinsinternes)

Protokoll der 2. Mitgliederversammlung von DANTE Deutschsprachige Anwendervereinigung T_EX e.V.

Luzia Dietsche

Zeit: 22. März 1990
 Beginn: 14.00 Uhr
 Ort: Heinrich-Heine-Universität
 Universitätsstr. 1
 Gebäude 2521 – HS 5E
 4000 Düsseldorf 1

Versammlungsleiter: Herr Joachim Lammarsch, 1. Vorsitzender
 Protokollantin: Frau Luzia Dietsche, Schriftführerin

Der Direktor des Rechenzentrums, Herr Prof. Dr. Jan Knop, hält eine kurze Begrüßungsansprache, in der er die Entwicklung und Anwendung von T_EX an der Heinrich-Heine-Universität beschrieb. Danach begrüßt der 1. Vorsitzende die Anwesenden, stellt sich und den Vorstand für die neuen Mitglieder vor und liest die mit der Einladung angekündigte, vorläufige Tagesordnung vor:

1. Wahl der Ehrenmitglieder.
2. T_EX 3.0/METAFONT 2.0 — und was nun?
3. DANTE e.V. — Vorschläge für die europäische Zusammenarbeit zur Diskussion während der GUT-Tagung?
4. DANTE e.V. — TUG, Vorschläge für das *board of directors* der TUG im Sommer auf der 11. T_EX-Tagung in Texas?
5. Softwareverteilung über Diskette, Listserver und Mailbox.
6. Vorstellung der Site-Koordinatoren.
7. Verschiedenes.

Nachdem auf die Frage nach weiteren Tagesordnungspunkten keine Meldungen erfolgten, geht Herr Lammarsch zum ersten Tagesordnungspunkt über.

zu Punkt 1:

Zur Wahl vorgeschlagen werden:

- Dr. Peter Sandner, Direktor des Rechenzentrums der Universität Heidelberg;

- Klaus Thull, Berlin.

Herr Dr. Sandner hat sich dank seiner Unterstützung zuerst der *deutschen T_EX-Interessenten* und dann von DANTE, Deutschsprachige Anwendervereinigung T_EX e.V., indirekt hohe Verdienste für die deutschsprachige T_EX-Gemeinde erworben. Ohne die moralischen und finanziellen Zuwendungen durch das Rechenzentrum der Universität Heidelberg und seiner Angestellten wäre der reibungslose Ablauf bei der Gründungsversammlung und der Aufbauphase des Vereins nicht denkbar gewesen.

Herr Thull hat dem Verein durch seine Implementation von T_EX auf IBM PCs, die er der Allgemeinheit als *public domain* Software zur Verfügung stellte, eine erste Basis gegeben. Dank dieser T_EX-Version hat der Verein die Grundlage für eine kontinuierliche Verteilung auf möglichst vielen Betriebssystemen erhalten. Aber nicht genug damit, hat Herr Thull auch ein METAFONT für PCs geschaffen, das er ebenfalls dem Verein zur Verfügung stellte.

Auf die Frage von Herrn Lammarsch, ob jemand Einwände gegen die Wahl der beiden Vorgeschlagenen zu Ehrenmitgliedern habe, erfolgten keine Meldungen. Damit wurden Herr Dr. Peter Sandner und Herr Klaus Thull zu Ehrenmitgliedern von DANTE, Deutschsprachige Anwendervereinigung T_EX e.V., ernannt. Die Präsente (je ein kalligraphischer Kalender, eine Ausgabe von Dante's *Göttlicher Komödie* und eine Ernennungsurkunde) konnten wegen der Abwesenheit der beiden neuen Mitglieder nicht überreicht werden.

Nachdem der erste Punkt der Tagesordnung abgehandelt war, schlägt Herr Lammarsch vor, die Reihenfolge der nächsten Themen zu verändern: Punkt 2 soll an das Ende der Versammlung verschoben werden, da zu erwarten ist, daß die Diskussion darüber länger dauern wird.

zu Punkt 5 und 7:

Unter *Verschiedenes* wird nun die Frage „Was hat sich bei DANTE e.V. getan?“ erörtert.

In der internationalen Zusammenarbeit mit anderen europäischen Gruppen wurde der Beschluß gefaßt, daß Gebühren bei Tagungen europäischer Vereinigungen für alle Mitglieder einer europäischen Vereinigung gleich hoch sein sollen (d.h. alle Mitglieder von DANTE e.V. bezahlen z.B. bei der französischsprachigen Tagung in Toulouse oder bei der niederländischen Tagung in Groningen dieselbe Tagungsgebühr wie Mitglieder von GUTenberg, bzw. NTG). Dieselbe Regelung gilt für die *proceedings* von Exeter (europäische T_EX-Tagung 1988): Mitglieder von DANTE e.V. bezahlen einen reduzierten Preis. Zu diesen *proce-*

dings werden die *abstracts* in LISTSERV@DHDURZ1 eingespielt, bzw. per Diskette von DANTE e.V. verteilt.

Auf dem Server in Heidelberg sind verschiedene neue Filelisten eingerichtet worden: eine GERMAN FILELIST, eine FRENCH FILELIST und eine RUSSIAN FILELIST, die jeweils sprachspezifische Stylefiles, Makros und Fonts enthalten. Außerdem wurde die neueste original L^AT_EX-Version von Leslie Lamport in die LATEX FILELIST eingespielt, die ab jetzt von Herrn Rainer Schoepf, Heidelberg, gewartet wird. Eine dem internationalen Standard angepaßte Version von L^AT_EX von Herrn Joachim Schrod, Rödermark, ist in der ILATEX FILELIST zu finden.

Die Umstrukturierung des Servers ist jetzt soweit fortgeschritten, daß ein Index zu den oben erwähnten Filelisten in der Master-Filelist TEX FILELIST zu finden ist.

Die vereinsinterne Kommunikations-Liste DANTE-L wird aufgelöst und alle hier Eingetragenen in die allgemeine TEX-D-L übernommen.

Auf dem Server werden in nächster Zeit die neue Version SBTEX 3.0 von Mr. Wayne Sullivan und T_EX für OS2 von Herrn André Kämmer eingespielt. Die genannte Software ist für Mitglieder ohne EARN-Zugang auch auf Diskette erhältlich. Ferner wird eine weitere T_EX-Version (EMTEX) für IBM PCs von Herrn Eberhard Mattes auf Diskette verteilt werden.

Es ist geplant, eine T_EX-Version für Atari ST von Herrn Christoph Strunk, Bochum, in die Verteilung zu nehmen. Herr Lammarsch steht außerdem mit Herrn Frank Wübbeling, Münster, in Verhandlung über eine *public domain* Version von T_EX für AMIGA.

Zum Server in Heidelberg wird ein neuer FTP-Server am Rechenzentrum der Universität in Stuttgart zur Verfügung gestellt, der von Frau Barbara Burr gewartet wird. Er enthält diverse T_EX-Software (z.B. EMTEX), die in Heidelberg nicht zur Verfügung steht. Es gibt dafür zwei Zugangsmöglichkeiten:

- | | | | |
|-----------|-------------------|----------|-------------|
| 1. telnet | 129.69.1.12 | 2. ftp | 129.69.1.12 |
| login | infoserv/softserv | login | ftp |
| password | guest | password | guest |
| | | cd | soft/tex |

Dabei ist zu beachten, daß in beiden Fällen von dem Rechner nur Kleinschreibung akzeptiert wird!

Herr Konrad Neuwirth, Wien, kündigt an, daß METAPLOT, ein Programm zum Einbinden von Vektorgrafiken auf IBM PCs und AMIGA, ab sofort als Shareware zum Preis von \$20 zur Verfügung steht.

Herr Lammarsch hat mit Addison-Wesley-Verlag Deutschland ausgehandelt, daß dessen Bücher über den Verein beziehbar sein werden. Da der Verein der Preisbindung von Buchhandlungen unterliegt, wird es für Mitglieder kei-

ne Ermäßigung geben. Allerdings werden die Einnahmen aus dem Verkauf der Bücher der Vereinskasse zugute kommen. Bücher aus den USA sind in diesem Angebot zur Zeit noch nicht enthalten; darüber laufen aber noch Verhandlungen.

Ein neues Makropaket von Michael Spivak, USA, mit L^AT_EX-ähnlichen *features* für $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX — L $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX — wird vom Verein an Interessierte zum Selbstkostenpreis weitergegeben.

Herr Lammarsch verschiebt eine Diskussion über „T_EX an Schulen“, ein bereits länger vorbereitetes Projekt, auf die Mitgliederversammlung im Herbst in Göttingen.

Fragen aus dem Forum:

Die Frage eines Anwesenden, ob die Möglichkeit besteht, sich von außerhalb in Heidelberg auf dem Rechner einzuloggen, um an die zur Verfügung stehende Software zu gelangen, muß verneint werden.

Ein zentraler Einkauf von Disketten von DANTE e.V. ist aus finanziellen und technischen Gründen nicht möglich.

zu Punkt 6:

Für die entschuldigten Site-Koordinatoren J. Schrod/Treiber und U. Untermarzoner/AIX berichtet Herr Lammarsch im Auftrag über die Entwicklungen.

Treiber — Joachim Schrod, Rödermark

Herr Schrod will in einer der nächsten Ausgaben der Komödie einen Fragebogen veröffentlichen, der eine Zusammenstellung der häufigsten Fragen und Antworten zu Treibern ermöglichen soll. Außerdem will er einen Test für Treiber entwickeln. *International L^AT_EX* wird von ihm weiter gewartet. Upgrades dazu werden über TIE-Files verteilt.

AIX — Uwe Untermarzoner, Tübingen

T_EX für AIX ist fertig. Es ist lauffähig auf IBM PS2 und RISC 6000. Im Paket sind Bildschirm- und Druckertreiber enthalten. Die Verteilung wird über DANTE e.V. laufen.

MVS — Ferdinand Hommes, St. Augustin

Die Treiber der Gesellschaft für Mathematische Datenverarbeitung (GMD) sind ab jetzt *public domain*, allerdings ohne Source-Code. Sie werden auf dem Server installiert. Das Amt des Site-Koordinators für MVS bleibt bei Herrn Hommes. Da die Aktivitäten der GMD in Bezug auf T_EX fast vollständig eingestellt wurden, wird es aber von dieser Seite aus keine Weiterentwicklung der Treiber mehr geben. Daher ist Herr Hommes auf der Suche nach jemandem, der bereit wäre, diese Aufgabe zu übernehmen.

- VM** — Herr Dr. Georg Bayer, Braunschweig
 Die Verteilung im VM bleibt dieselbe wie bisher.
 Um etwaigen Mißverständnis vorzubeugen, betont Herr Lammarsch, daß der Site-Koordinator im deutschsprachigen Raum Herr Dr. Bayer ist. Sein Amt als Site-Koordinator der T_EX Users Group (TUG) für VM beinhaltet eine völlig andere Aufgabenstellung und Zielrichtung.
- UNIX** — Karl Kleine, Karlsruhe
 Herr Kleine ist nicht anwesend.
- BS 2000** — Friedhelm Sowa, Düsseldorf
 Herr Sowa hat einen Treiber für Calcomp-Plotter entwickelt, der sich noch in der Test-Phase befindet.
- Atari** — Lutz Birkhahn, Cadolzburg/Stefan Lindner, Fürth
 Die Herren Birkhahn/Lindner sind mit der Bereitstellung eines postscript-fähigen Druckertreibers fast fertig. Ihre T_EX-Version, die bisher als Shareware zur Verfügung steht, soll demnächst für Mitglieder freigegeben werden.
 Den beiden Koordinatoren werden sehr häufig Fragen nach allgemeiner Unterstützung gestellt, was zu der Feststellung führt, daß die Existenz von DANTE e.V. noch immer viel zu wenig bekannt ist. Dieser Meinung schließt sich Herr Lammarsch an und ruft die Anwesenden dazu auf, mehr in Zeitschriften über T_EX und DANTE e.V. zu veröffentlichen. Er selber hat schon versucht, einzelne Artikel unterzubringen, bisher jedoch ohne Erfolg. Es stellt sich die Frage, ob das mit andersgelagerten finanziellen Interessen der Magazine zu tun haben kann.
- AMIGA** — Konrad Neuwirth, Wien
 Eine Liste von verfügbarer Software für AMIGA-T_EX soll demnächst fertiggestellt werden. Die Verteilung wird in der üblichen Weise erfolgen.
- NOS/VSE** — Norbert Schwarz, Bochum
 Für NOS/VSE gibt es nichts Besonderes zu berichten.
- Macintosh** — Lothar Meyer-Lerbs, Bremen
 Es ist geplant, die *public domain* Version OZTEX von Aston zu besorgen und zu testen.
- METAFONT** — Norbert Schwarz, Bochum
 Auf amerikanischer Seite wurden russische Schriftzeichen und Trennmuster entwickelt, die von verschiedenen Servern und von DANTE e.V. auf Diskette erhältlich sind.
- L^AT_EX-Style-Files** — Dr. Hubert Partl, Wien
 Die Pflege von GERMAN STY/TEX bleibt bis zum endgültigen Wechsel zu T_EX 3.0 bei Herrn Dr. Partl und geht dann zu Herrn Mittelbach, Mainz, über, der auch für das neue L^AT_EX verantwortlich sein wird. Es muß je-

doch gewährleistet bleiben, daß die Option GERMAN für plain T_EX ebenso weiterhin unterstützt wird.

zu Punkt 3 und 4:

Als Einstieg erläutert Herr Lammarsch die Vorschläge des französischsprachigen Vereins zur Tagesordnung in Toulouse (Mai 1990).

Bernard Gaulle (Präsident von GUTenberg, Groupe Utilisateur de T_EX) möchte als erstes die Notwendigkeit von *vice presidents* in der TUG diskutieren, bzw. die Frage, ob sie nicht eine Papiertiger- oder Alibifunktion haben.

Die nächsten Fragen drehen sich um eine bessere und schnellere Verteilung von TUGboat innerhalb Europas, die Verteilung von T_EXhax über Listserv in Heidelberg, die Koordination und Weiterentwicklung von T_EX 3.0 und L^AT_EX 2.10, und in diesem Zusammenhang insbesondere die Frage nach europäischen Fonts und internationalen Style-Files.

Anregungen aus dem Forum:

Die Mehrsprachigkeit für alle Pakete (nicht nur für L^AT_EX) muß gewährleistet werden. Europäische Fonts sind sehr wichtig für alle (außer für Amerikaner), sollten jedoch nicht gegen den Willen der TUG durchgefochten werden. Deutschsprachige Texte enthalten mit Abstand die meisten Großbuchstaben: diese sind im Verhältnis zu den Kleinbuchstaben zu massiv und sollten abgemagert werden.

Die Frage, wie die Bestellung von Büchern und die Bezahlung von Mitgliedsbeiträgen für Mitglieder von DANTE e.V. bei der TUG angenehmer gestaltet werden kann, wird von Herrn Lammarsch in Toulouse, bzw. College Station erneut angesprochen werden.

Eine Verteilung von TUGboat über eine europäische Stelle ist in nächster Zukunft nicht in Sicht. Herr Lammarsch will vorschlagen, die Mitgliedschaft vom Bezug der Zeitung zu trennen, so daß ohne erhöhte Kosten mehrere Exemplare bestellt werden können.

Diesen Anregungen aus dem Forum folgte eine Kaffeepause, in deren Anschluß schließlich zu Punkt 2 der Tagesordnung übergegangen wurde.

zu Punkt 2:

Da sich Herr Peter Breitenlohner, München, bereits ausführlicher mit dem Thema T_EX 3.0 beschäftigt hat, berichtet er zu Anfang darüber, wie er sich eine Verwendung von T_EX 3.0 in einer mehrsprachigen Umgebung vorstellen könnte. Er hat dazu verschiedene Makropakete entwickelt, die er mit WEBMAC getestet hat. Die folgende Forumsdiskussion dreht sich im wesentlichen um die Frage

der Eingabe, der internen Darstellung der Code-Tabelle und der für das neue T_EX benötigten Fonts. Ergebnisse an diesem Tag waren:

- Eine Anpassung darf nicht nur die Sprache selbst betreffen (z.B. die Umwandlung von *Chapter* in *Kapitel*), sie muß auch Abstände, Trennung u.ä. berücksichtigen.
- Eine bessere Zusammenarbeit der internationalen Site-Koordinatoren wird gefordert.
- Herr Erich Neuwirth, Wien, will die Code-Tabelle, die an der Washington-State-University bereits entwickelt wird, an Interessierte verschicken.
- Wichtig bei der Verwendung des neuen T_EX ist vor allem, wie die interne Code-Tabelle und die Belegung der Zeichensätze festgelegt wird. Wenn die Tabelle, bzw. die Belegung einmal feststeht, muß sie aus Kompatibilitätsgründen auch so beibehalten werden.

Über diese Fragen konnte kein Konsens gefunden werden. Es gab darüber zwei Meinungen:

1. Die ersten 128 Zeichen der Tabelle sollen so bleiben, wie sie sind. Nur die zweite Hälfte der Tabelle wird erneuert.
2. Alle 256 Zeichen sollen erneuert und damit gleichzeitig an die ISO-Norm angepaßt werden, damit sie internationalem Standard entspricht.

Da in diesem Punkt keine Einigung erzielt werden konnte, schlägt Herr Lamarsch vor, die Diskussion am nächsten Tag fortzuführen. Nach den Kursen am 23.3.90 fand sich deshalb erneut eine Gruppe Interessierter zusammen. Die Ergebnisse dieses Tages waren:

- Deutsch- und französischsprachige Sonderzeichen sollten in einem Aufwasch gemacht werden. Dies empfiehlt sich auch wegen des guten Schlußes mit den Franzosen.
- Es soll eine Einigung mit GUTenberg über einen Fontssatz erzielt werden, mit dem getestet werden kann. Daran können sich weitere Änderungen anschließen.
- Ein Umsetzungsprogramm von 128 in 256 Zeichen für Treiber muß geschrieben werden.
- PubliCMETAFONT muß auf die neue Version angepaßt werden.
- Ein vorläufiger Name für einen neuen europäischen Font könnte *ac* (für *almost computer*) lauten. Ein endgültiger Name könnte *ec* (für *european computer*) sein.

- GERMAN muß auch weiterhin funktionieren.
- Herr Breitenlohner macht einen Vorschlag für eine Anpassung deutschsprachiger Makrotexe als *workable example* fertig (Entwurf).
- Herr Schwarz entwickelt Trennmuster und einen neuen Font mit 256 Zeichen (`acmr10`), in dem die deutschsprachigen Sonderzeichen vorhanden sind. Der Rest der zusätzlichen 128 Zeichen soll in diesem Entwurf vorerst leerbleiben.

Mit diesen Ergebnissen beendet Herr Lammarsh mit einem Dank an alle Anwesenden und Helfer endgültig die Versammlung.

Luzia Dietsche
(Schriftführerin)

Von fremden Bühnen

Deutsche Kurzfassungen: TUGboat-Artikel Vol. 11, No. 1

Luzia Dietsche

Virtuelle Fonts: Noch mehr Spaß für Genies (D. E. Knuth)

Das größte Genie der T_EX-Welt erklärt allen anderen, sprich den T_EX-Implementatoren, wie man fremde Fonts problemlos in T_EX einarbeiten kann. Dazu stellt er ein neues Programm, VFtoVP und VPtoVF, vor, das mit TFtoPL und PLtoTF vergleichbar ist. Ein kurzer Ausschnitt des Programms befindet sich am Anschluß an den Artikel.

Internationales L^AT_EX fertig zum Gebrauch (J. Schrod)

Internationales L^AT_EX (kurz I^AT_EX) ist eine freie und gewartete Version von L^AT_EX, mit der es möglich ist, auch nicht-englische Dokumente mit dem Standard-Layout von L^AT_EX zu setzen. Zusätzlich wurde ein bisher vorhandenes Problem mit der Auswahl der Größe von mathematischen Begrenzern gelöst. Diese Lösung kann auch für andere Makropakete verwendet werden, die jedoch auf plain-T_EX basieren müssen.

Unübliche Formen bei Absätzen (V. Eickhout)

Der Autor stellt drei interessante Möglichkeiten des Zeilen- und Absatzumbruchs vor. Er verwendet dazu die T_EX-Kommandos `\everypar` und `\lastbox`. Die von ihm gezeigten Makrolösungen ermöglichen es unter anderem, Text einzugeben, auch ohne die für die Formatierung nötigen Kommandos dazwischen zu schreiben.

Eine vollständige *public domain* Version von T_EX für IBM PCs und kompatible: die „zwei GUT Disketten“ (N. Brouard)

GUTenberg, der Verein für französischsprachige T_EX-Benutzer, hat auf zwei 1.2 MByte-Disketten eine *public domain* Version von T_EX und diversen Tools für MSDOS zusammengestellt. Als Tools sind dabei ein Texteditor, Bildschirm- und Druckertreiber. Die T_EX-Version (SB26TEX) enthält ein komplettes L^AT_EX sowie BIBT_EX 0.99c. Als Bildschirmtreiber ist CDVI 1.0, als Druckertreiber die Treiberfamilie DVlxxx dabei. Außerdem wurde einiges an Dokumentation in französisch und englisch hinzugefügt. Der Artikel selbst ist in französischer Sprache geschrieben.

Auflisten von mathematischen Makros (M. J. Wichura)

Oft hat man in einem mit \TeX formatierten Dokument Makros für immer wiederkehrende mathematische Ausdrücke definiert und weiß bei Bedarf nicht mehr, wie die Definition aussah. Dafür stellt der Autor eine Möglichkeit vor, solche eigenen Definitionen in einem extra File auflisten zu lassen. Allerdings ist `\ListMacrosOnFile` nur im mathematischen Modus anwendbar. Definitionen außerhalb werden nicht akzeptiert. Eine ausführliche Beschreibung, wie `\ListMacrosOnFile` aufgebaut ist, findet sich im zweiten Teil des Artikels.

Beispiele und Anwendungstechniken für Output Routinen.

Teil I: Einführung und Beispiele (D. Salomon)

Dieser Artikel ist der 1. Teil einer 3-teiligen Serie über Output Routinen von \TeX . Teil 1 gibt eine anschauliche Einführung mit einfachen Beispielen. Teil 2 wird Markierungen, speziellen `penalty`-Werten, `kerns` und `extra` in den Text eingefügten Boxen gewidmet sein. Teil 3 wird sich schließlich mit Einfügungen beschäftigen. Im vorliegenden Teil definiert D. Salomon einführend Begriffe wie *Box*, *List*, *Main Loop*, *Main Vertical List* und *Page Builder*. Er erklärt, wie, wann, warum und von welchem Makro die Ausgabe einer Seite aufgerufen wird. An konkreten Beispielen (mit Listing) zeigt er, wie `\output` variiert und an verschiedene Bedürfnisse angepaßt werden kann.

Querverweise und der letzte „Dirty Trick“ (L. K. Durst)

Hier führt L. Durst sein Tutorial von TUGboat, Vol. 10, No. 3 fort. Er beschreibt, wie man ein „feature“ von D. E. Knuth, das dieser zur Syntaxüberprüfung entwickelt hat, auch auf andere Weise nützlich anwenden kann. Er verwendet dieses „Syntax Checking“, um verschiedene Informationen, z.B. für Querverweise, in einem Vorlauf in einen externen File schreiben zu lassen. Dieser File wird dann beim tatsächlichen \TeX -Durchlauf gelesen und verarbeitet. Auf diese Art und Weise ist es möglich, zwei komplette Durchläufe für Verweise, Indexeinträge etc. auf 1 1/2 zu verkürzen.

Der Gebrauch von \TeX 3 in einer mehrsprachigen Umgebung — Einige Vorschläge (P. Breitenlohner)

Wie soll das neue \TeX angewandt werden, so daß es zuerst einmal portabel, dann benutzerfreundlich und zu guter Letzt auch noch effizient bleibt? Dieser Frage geht P. Breitenlohner nach und entwirft ein grobes Schema, durch das seiner Meinung nach die drei genannten Kriterien in der vorgegebenen Reihenfolge verwirklicht werden könnten. Er sieht das Schema nicht als zwingend, sondern als Diskussions-Vorschlag.

Wie man es vermeidet, lange Sätze in T_EX's \write-Ausgabestrom zu schreiben (P. Breitenlohner)

Verweisend auf einen Artikel von D. Salomon in TUGboat Vol. 10, No. 3 stellt P. Breitenlohner ein (plain) T_EX Makro vor, mit dem man es vermeiden kann, daß die Sätze einer Überschrift zu lange werden. Er splittet den Output der Überschrift in mehrere Teile auf und zwar jeweils an der Stelle, wo in der Eingabe eine neue Zeile beginnt.

Druckfehler: Die neue Version von T_EX und METAFONT, TUGboat Vol. 10, No. 3 (D. E. Knuth)

Dieser knappe Artikel enthält eine Korrektur von den Autoren, die einen Absatz über das neue T_EX und METAFONT auf Seite 326, Spalte 1 ersetzen soll.

T_EX, TUG und Ost-Europa (H. Partl)

H. Partl berichtet über eine Tagung in der Tschechoslowakei, bei der erstaunt feststellen konnte, wie weit T_EX in Osteuropa verbreitet ist. Die Bereitschaft, Kontakte zu knüpfen, ist in diesen (zu jener Zeit noch unfreien) Staaten enorm hoch, die Entwicklung und Anpassung an die nationalen Bedürfnisse weit gediehen.

**Richtlinien für das Erstellen eines Journal-Style:
Ein Bericht über einen neuen L^AT_EX-Style (K. v. d. Laan)**

Dieser Bericht ist eine Ausarbeitung darüber, wie man den allgemeinen `article style` an spezielle Journale anpassen kann. Das nächste Ziel der Autoren ist es, generische Richtlinien für das Erstellen von Journal-Styles zu entwickeln.

Koordination von nicht-englischen Anwendungen von T_EX (M. Ferguson)

M. Ferguson stellt sich hier als Koordinator für nicht-englische Anwendungen von T_EX vor. Zuerst bot er sich als Freiwilliger zum Sammeln von Trenn-Mustern an, erweiterte dies aber dann auf spezielle Makros, Stylefiles, Fonts u.a.m. Außer dem berichtet er in dem Artikel kurz, wie er sich eine Anpassung der nicht-englischen Besonderheiten an T_EX 3 vorstellt.

Die L^AT_EX-Ecke (J. Damrau)

In dieser Ecke vom TUGboat will J. Damrau ab jetzt wieder Tips und Tricks für L^AT_EX-Neulinge vorstellen. Dieser Beitrag ist dem Problem gewidmet, wie L^AT_EX zu Beginn oder auch mitten im Text zu doppeltem Zeilenabstand überredet werden kann.

Eine Stil-Option, um Standard-L^AT_EX-Dokumentstyles an das A4-Maß anzupassen (N. Poppelier und J. Braams)

Die meisten L^AT_EX-Benutzer aus europäischen Ländern werden sich schon einmal über die ganz und gar amerikanische Seitenaufteilung von L. Lamport geärgert haben, die dem in Europa verbreiteten A4-Maß nicht im geringsten gerecht wird. Die beiden Autoren haben deshalb eine bereits vorhandene Stil-Option weiterentwickelt, so daß jetzt u.a. auch Deckungsgleichheit bei zweiseitigem Drucken, genügend linker Rand zum Binden und genügend rechten Rand für Randnotizen übrigbleibt.

***T_EX*T1 wird public domain (D. Guenther)**

Die Washington State University hat Anfang Januar verschiedene Makropakete freigegeben, die bislang gekauft werden mußten. Dabei ist *T_EX*T1, ein Paket, das wie L^AT_EX auf plain T_EX aufsetzt und ähnliche Möglichkeiten bietet. Laut D. Guenther ist *T_EX*T1 leichter zu modifizieren als L^AT_EX. Weitere freigegebene Produkte sind ein Compugraphics 8600 Fotosatz Dvi-Treiber, zusätzliche Fonts aus der Computer Modern Familie in den Point-Größen 11, 12, 14, 18, 24 und 36 und ein Font für das Internationale Phonetische Alphabet in den Point-Größen 9, 10, 11 und 12.

Das neue Auswahlschema für Fontfamilien — Eine Benutzerschnittstelle zu Standard-L^AT_EX (F. Mittelbach und R. Schöpf)

Die beiden Autoren beschreiben in Anlehnung an einen von ihnen veröffentlichten Artikel in TUGboat Vol. 10, No. 3, wie das neue Auswahlschema für Fonts in der Standard-L^AT_EX-Umgebung verwendet werden kann. Hauptmerkmale dieses Schemas sind die Möglichkeit, Familien, Reihen, Formen und Größen unabhängig voneinander wechseln zu können, ein Style-File, mit dem auch alte Dokumente ohne Veränderung ihres Aussehens und ihrer Eingabe bearbeitet werden können, und ein Makro-Zusatzpaket, das kompatibel zu bereits vorhandenen Standard-Dokument-Stylefiles ist.

Die Bretter, die die Welt bedeuten

$\text{T}_{\text{E}}\text{X}$ nisches

Im folgenden Donald E. Knuths Ausführungen zu den Änderungen in $\text{T}_{\text{E}}\text{X}$ 3.0 im Original:

The New Versions of $\text{T}_{\text{E}}\text{X}$ and METAFONT



Donald E. Knuth

For more than five years I held firm to my conviction that a stable system was far better than a system that continues to evolve. But during the TUG meeting at Stanford in August, 1989, I was persuaded to make one last set of changes, in order to bring $\text{T}_{\text{E}}\text{X}$ and METAFONT to a state of completion consistent with their overall philosophy and goals.

The main reason for the changes was the fact that I had guessed wrong about 7-bit character sets versus 8-bit character sets. I believed that standard text input would continue indefinitely to be confined to at most 128 characters, since I did not think a keyboard with 256 different outputs would be especially efficient. Needless to say, I was proved wrong, especially by developments in Europe and Asia. As soon as I realized that a text formatting program with 7-bit input would rapidly begin to seem as archaic as the 6-bit systems we once had, I knew that a fundamental revision was necessary.

But the 7-bit assumption pervaded everything, so I needed to take the programs apart and redo them thoroughly in 8-bit style. This put $\text{T}_{\text{E}}\text{X}$ onto the operating table and under the knife for the first time since 1984, and I had a final opportunity to include a few new features that had occurred to me or been suggested by users since then.

The new extensions are entirely upward compatible with previous versions of $\text{T}_{\text{E}}\text{X}$ and METAFONT (with a few small exceptions mentioned below). This means that error-free inputs to the old $\text{T}_{\text{E}}\text{X}$ and METAFONT will still be error-free inputs to the new systems, and they will still produce the same outputs.

However, anybody who dares to use the new extensions will be unable to get the desired results from old versions of $\text{T}_{\text{E}}\text{X}$ and METAFONT. I am therefore asking the $\text{T}_{\text{E}}\text{X}$ community to update all copies of the old versions as soon as possible. Let us root out and destroy the obsolete 7-bit systems, even though we were able to do many fine things with them.

In this note I'll discuss the changes, one by one; then I'll describe the exceptions to upward compatibility.

1. The character set. Up to 256 distinct characters are now allowed in input files. The codes that were formerly limited to the range 0..127 are now in the range 0..255. All characters are alike; you are free to use any character for any purpose in \TeX , assigning appropriate values to its `\catcode`, `\mathcode`, `\lccode`, `\uccode`, `\sfcode`, and `\delcode`. Plain \TeX initializes these code values for characters above 127 just as it initializes the codes for ordinary punctuation characters like ‘!’.

There’s a new convention for inputting an arbitrary 8-bit character to \TeX when you can’t necessarily type it: The four consecutive characters $\text{\~}\alpha\beta$, where α and β are any of the “lowercase hexadecimal digits” 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, or f, are treated by \TeX on input as if they were a single character with specified code digits. For example, $\text{\~}80$ gives character code 128; the entire character set is available from $\text{\~}00$ to $\text{\~}ff$. The old convention discussed in Appendix C, under which character 0 was $\text{\~}0$, character 1 (control-A) was $\text{\~}A$, . . . , and character 127 was $\text{\~}?$, still works for the first 128 character codes, except that the character following \~ should not be a lowercase hexadecimal digit when the immediately following character is another such digit.

The existence of 8-bit characters has less effect in METAFONT than in \TeX , because METAFONT’s character classes are built in to each installation. The normal set of 95 printing characters described on page 51 of *The METAFONTbook* can be supplemented by extended characters as discussed on page 282, but this is rarely done because it leads to problems of portability. METAFONT’s `char` operator is now redefined to operate modulo 256 instead of modulo 128.

2. Hyphenation tables. Up to 256 distinct sets of rules for hyphenation are now allowed in \TeX . There’s a new integer parameter called `\language`, whose current value specifies the hyphenation convention in force. If `\language` is negative or greater than 255, \TeX acts as if `\language = 0`.

When you list hyphenation exceptions with \TeX ’s `\hyphenation` primitive, those exceptions apply to the current language only. Similarly, the `\patterns` primitive tells \TeX to remember new hyphenation patterns for the current language; this operation is allowed only in the special “initialization” program called INITEX. Hyphenation exceptions can be added at any time, but new patterns cannot be added after a paragraph has been typeset.

When \TeX reads the text of a paragraph, it automatically inserts “whatsit nodes” into the horizontal list for that paragraph whenever a character comes from a different `\language` than its predecessor. In that way \TeX can tell what hyphenation rules to use on each word of the paragraph even if you switch frequently back and forth among many different languages.

The special `whatsit` nodes are inserted automatically in unrestricted horizontal mode (i.e., when you are creating a paragraph, but not when you are specifying the contents of an `hbox`). You can insert a special `whatsit` yourself in restricted horizontal mode by saying `\language⟨number⟩`. This is needed only if you are doing something tricky, like unboxing some contribution to a paragraph.

3. Hyphenated fragment control. T_EX has new parameters `\lefthyphenmin` and `\righthyphenmin`, which specify the smallest word fragments that will appear at the beginning or end of a word that has been hyphenated. Previously the values `\lefthyphenmin=2` and `\righthyphenmin=3` were hard-wired into T_EX and impossible to change. Now plain T_EX format supplies the old values, which are still recommended for most American publications; but you can get more hyphens by decreasing these parameters, and you can get fewer hyphens by increasing them. If the sum of `\lefthyphenmin` and `\righthyphenmin` is 63 or more, all hyphenation is suppressed. (You can also suppress hyphenation by using a font with `\hyphenchar=-1`, or by switching to a `\language` that has no hyphenation patterns or exceptions.)

4. Smarter ligatures. Now here's the most radical change. Previous versions of T_EX had only one kind of ligature, in which two characters like 'f' and 'i' were changed into a single character like 'fi' when they appeared consecutively. The new T_EX understands much more complex constructions by which, for example, we could change an 'i' following 'f' to a dotless 'i' while the 'f' remains unchanged: 'fi'.

As before, you get ligatures only if they have been provided in the font you are using. So let's look at the new features of METAFONT by which enhanced ligatures can be created. A METAFONT programmer can specify a "ligature/kerning program" for any character of the font being created. If, for example, the 'fi' combination appears in font position 12, the replacement of 'f' and 'i' by 'fi' is specified by including the statement

```
"i" =: 12
```

in the ligature/kerning program for "f"; this is METAFONT's present convention.

The new ligatures allow you to retain one or both of the original characters while inserting a new one. Instead of `=:` you can also write `|=:` if you wish to retain the left character, or `=:|` if you wish to retain the right character, or `|=:|` if you want to keep them both. For example, if the dotless `ı` appears in

font position 16, you can get the behavior mentioned above by having

```
"i" |=: 16
```

in f's program.

There also are four additional operators

```
|=:>,    =:|>,    |=:|>,    |=:|>>,
```

where each > tells T_EX to shift its focus one position to the right. For example, if f and i had been replaced by f and dotless ı as above, T_EX would begin again to execute f's ligature/kern program, possibly inserting a kern before the dotless ı, or possibly changing the f to an entirely different character, etc. But if the instruction had been

```
"i" |=:> 16
```

instead, T_EX would turn immediately to the ligature/kern program for characters following character 16 (the dotless ı); no further change would be made between f and ı even if the font had something specified there.

5. Boundary ligatures. Every consecutive string of ‘characters’ read by T_EX in horizontal mode (after macro expansion) can be called a ‘word’. (Technically we consider a ‘character’ in this definition to be either a character whose `\catcode` is a letter or otherchar, or a control sequence that has been `\let` equal to such a character, or a control sequence that has been defined by `\chardef`, or the construction `\char(number)`.) The new T_EX now imagines that there is an invisible “left boundary character” just before every such word, and an invisible “right boundary character” just after it. These boundary characters take effect if the font designer has specified ligatures and/or kerning between them and the adjacent letters. Thus, the first or last character of a word can now be made to change its shape automatically.

A ligature/kern program for the left boundary character is specified within METAFONT by using the special label `||:` in a **ligtable** command. A ligature or kern with the right boundary character is specified by assigning a value to the new internal METAFONT parameter *boundarychar*, and by specifying a ligature or kern with respect to this character. The *boundarychar* may or may not exist as a real character in the font.

For example, suppose we want to change the first letter of a word from ‘F’ to ‘ff’ if we are doing some olde English. The METAFONT font designer could then say

```
ligtable ||: "F" |=: 11
```

if character 11 is the ‘ff’. The same ligtable instruction should appear in the programs for characters like (and ‘ and “ and - that can precede strings of letters; then ‘Bassington-French’ will yield ‘Bassington-ffrench’.

If the ‘s’ of our font is the pre-19th century s that looks like a mutilated ‘f’, and if we have a modern ‘s’ in position 128, we can convert the final s’s as Ben Franklin did by introducing ligature instructions such as

```
boundarychar := 255;
ligtable "s": 255 =:| 128,
              "." =:| 128,
              ", " =:| 128,
              ")" =:| 128,
              "' " =:| 128,
```

and so on. (A true oldstyle font would also have ligatures for ss and si and sl and ssi and ssl and st; it would be fun to create a Computer Modern Oldstyle.)

The implicit left boundary character is omitted by \TeX if you say `\noboundary` just before the word; the implicit right boundary is omitted if you say `\noboundary` just after it.

6. More compact ligatures. Two or more ligtables can now share common code. To do this in METAFONT, you say ‘`skipto <n>`’ at the end of one `ligtable` command, then you say ‘`<n>::`’ within another. Such local labels can be reused; e.g., you can say `skipto 1` again after `1::` has appeared, and this skips to the *next* appearance of `1::`. There are 256 local labels, numbered 0 to 255. Restriction: At most 128 ligature or kern commands can intervene between a `skipto` and its matching label.

The TFM file format has been upwardly extended to allow more than 32,500 ligature/kern commands per font. (Previously there was an effective limit of 256.)

7. Better looking sloppiness. There is now a better way to avoid overfull boxes, for people who don’t want to look at their documents to fix unfeasible line breaks manually. Previously people tried to do this by setting `\tolerance=10000`, but the result was terrible because \TeX would tend to consolidate all the badness in one truly horrible line. (\TeX considers all badness ≥ 10000 to be infinitely bad, and all these infinities are equal.)

The new feature is a dimension parameter called `\emergencystretch`. If `\emergencystretch` is positive and if \TeX has been unable to typeset a paragraph without exceeding the given tolerances, another pass over the paragraph is made in which \TeX pretends that additional stretchability equal to

`\emergencystretch` is present in every line. The effect of this is to scale down all the badnesses into a range where previously infinite cases become finite; T_EX will find an optimum solution to the scaled-down problem, and this will be about as good as possible in a practical sense. (The extra stretching is not really present; therefore underfull boxes will be reported in warning messages unless `\hbadness` is increased.)

8. Looking at badness. T_EX has a new internal integer parameter called `\badness` that records the badness of the box it has most recently constructed. If that box was overfull, `\badness` will be 1000000; otherwise `\badness` will be between 0 and 10000.

9. Looking at the line number. T_EX also has a new internal integer parameter called `\inputlineno`, which contains the number of the line that T_EX would show on an error message if an error occurred now. (This parameter and `\badness` are “read only” in the same way as `\lastpenalty`: You can use them in the context of a `<number>`, e.g., by saying ‘`\ifnum\inputlineno>\badness ... \fi`’ or ‘`\the\inputlineno`’, but you cannot set them to new values.)

10. Not looking at error context. There’s a new integer parameter called `\errorcontextlines` that specifies the maximum number of two-line pairs of context displayed with T_EX’s error messages (in addition to the top and bottom lines, which always appear). Plain T_EX now sets `\errorcontextlines=5`, but higher level format packages might prefer `\errorcontextlines=1` or even `\errorcontextlines=0`. In the latter case, an error that previously involved three or more pairs of context would now appear as follows:

```
! Error.
(somewhere) The \top
                line
... 1.123 \The
                bottom line.
```

(If `\errorcontextlines<0` you wouldn’t even see the ‘...’ here.)

11. Output recycling. One more new integer parameter completes the set. If `\holdinginserts>0` when T_EX is putting the current page into `\box255` for the `\output` routine, T_EX will not move anything from insertion nodes into the corresponding boxes; all insertion nodes will stay in place. Designers of output routines can use this when they want to put the contents of box 255 back into the current page to be re-broken (because they might want to change `\size` or something).

12. Exceptions to upward compatibility. The new features of *T_EX* and METAFONT imply that a few things work differently than before. I will try to list all such cases here (except when the previous behavior was erroneous due to a bug in *T_EX* or METAFONT). I don't know of any cases where users will actually be affected, because all of these exceptions are pretty esoteric.

- *T_EX* used to convert the character strings `^^0`, `^^1`, ..., `^^9`, `^^a`, `^^b`, `^^c`, `^^d`, `^^e`, `^^f` into the respective single characters `p`, `q`, ..., `y`, `!`, `"`, `#`, `$`, `%`, `&`. It will no longer do this if the following character is one of the characters `0123456789abcdef`.

- *T_EX* used to insert no character at the end of an input line if `\endlinechar>127`. It will now insert a character unless `\endlinechar>255`. (As previously, `\endlinechar<0` suppresses the end-of-line character. This character is normally `13 = ASCII control-M = carriage return`.)

- Some diagnostic messages from *T_EX* used to have the notation `["80] ... ["FF]` when referring to characters `128...255` (for example when displaying the contents of an overflow box involving fonts that include such characters). The notation `^^80 ... ^^ff` is now used instead.

- The expressions `char128` and `char0` used to be equivalent in METAFONT; now `char` is defined modulo 256 instead. Hence `char-1 = char255`, etc.

- *INITEX* used to forget all previous hyphenation patterns each time you specified `\patterns`. Now all hyphenation pattern specifications are cumulative, and you are not permitted to use `\patterns` after a paragraph has been hyphenated by *INITEX*.

- *T_EX* used to act a bit differently when you tried to typeset missing characters of a font. A missing character is now considered to be a word boundary, so you will get slightly more diagnostic output when `\tracingcommands>0`.

- *T_EX* and METAFONT will report different statistics at the end of a run because they now have a different number of primitives.

- Programs that use the string pool feature of TANGLE will no longer run without changes, because the new TANGLE starts numbering multicharacter strings at 256 instead of 128.

- *INITEX* programs must now set `\lefthyphenmin=2` and `\righthyphenmin=3` in order to reproduce their previous behavior.

BIB_TE_X und MAKEINDEX

Gabriele Kruljac

Auf der europäischen T_EX-Tagung im September 1989 in Karlsruhe fand unter anderem eine Gruppensitzung zum Thema BIB_TE_X und MAKEINDEX statt.

Zu MAKEINDEX gab es einige Anregungen, um die sich Herr Peter Schilling weiter kümmern wollte.

Die verschiedenen gesammelten Anforderungen und Wünsche zu BIB_TE_X hat Herr Wonneberger im Januar nochmals über die T_EX-EURO-Liste per e-mail bekannt gemacht. Damit auch unsere nicht-vernetzten Mitglieder diese Informationen erhalten, hier eine kurze übersetzte Zusammenfassung:

Insgesamt war man sich einig, daß BIB_TE_X ein wertvolles Instrument zur Erzeugung von Literaturlisten sein kann, nicht nur im Bereich der Hochschulen, sondern auch in der Industrie. So wäre z.B. eine Verteilung von .bib-Files für neue Bücher oder Artikel denkbar.

Die Anregungen im einzelnen waren:

- Möglichkeit von Rückwärts-Referenzen von Gesamtwerken auf einzelne Teile; damit könnte alles in einem Durchlauf korrekt erstellt werden und die Konsistenz bliebe gewährleistet.
- Definition von Unterstyles (wie mit dem `\documentstyle` Befehl), die die Übernahme einzelner Felder in die Liste steuern; der Haupt-Style sollte nur das Labeln und Sortieren übernehmen.
- Sprach-Flag, der sowohl die Texte („Auflage“ oder „Edition“) setzt, wie auch z.B. im Deutschen die Kleinschreibung von Titeln verhindert.
- Vorgabe für die Formatierung der Schlüssel der Verweise.
- Unter-Styles, die diese Schlüssel sichtbar machen.
- Eingabe mehrerer Schlüssel oder Alias-Felder, um verschiedene Arten der Bezugnahme zuzulassen.
- Ein L^AT_EX `\addcommand`, um Kommandos einzufügen, die Bestandteil des Dokuments sein können oder auch nicht.
- Neue Einträge: JOURNAL und YEARBOOK.
- Felder für ISBN, ISSN oder generelle Nummer.
- Neues Feld für Kommentar und Schlagworte.
- Kombinationsmöglichkeiten wie z.B.: 1. Aufl. 1980, 2. Aufl. 1982: Band I: 1989, II: 1984.
- Behandlung von Titeln im Namensfeld („Dr.“ ist im Deutschen Bestandteil des Namens).

- Verteilung von BIBTEX-Einträgen über die TUG (TeXhax, ...), zumindest von TEX-Literatur, TUGboat, ...
- Weiterentwicklungen von BIBTEX sollten Standards wie RAK berücksichtigen.
- Bibliographische Einträge durch BIBTEX nacheinander, z.B. in den Fußnoten eines Dokumentes.

Frau Anne Brüggemann-Klein hat sich bereit erklärt, die Liste der Wünsche zusammenzustellen und weiterzuleiten.

Setzen von Regeln mit `regel.sty`

Thomas Birnthaler

Für: L^AT_EX.

Aufgabe: Stellt eine Umgebung zum Setzen von Regeln zur Verfügung.

Eine übliche Regel hat etwa die Form

```

IF      (Bedingung 1
        AND ...
        AND Bedingung N)
THEN   (Folgerung 1
        OR ...
        OR Folgerung M)

```

(1)

und wurde hier mit der `regel`-Umgebung durch folgende Eingabe erzeugt:

```

\begin{regel}
  \IF{Bedingung 1 \AND \ldots \AND Bedingung N}
  \THEN{Folgerung 1 \OR \ldots \OR Folgerung M}
  \label{regel1}
\end{regel}

```

Regeln werden in den vorhergehenden bzw. folgenden Absatz mit aufgenommen, wenn sie oben bzw. unten *nicht* durch eine Leerzeile von diesem abgetrennt sind. Regel (1) wurde z.B. ohne Leerzeilen zwischen Text und `regel`-Umgebung in den umgebenden Absatz eingebaut, während die folgende Regel (2) durch Leerzeilen getrennt einen Absatz für sich allein bildet.

```

IF      (Bedingung 1
        AND ...
        AND Bedingung N)
THEN   (Folgerung 1
        OR ...
        OR Folgerung M)

```

(2)

In den folgenden Beispielen für Regeln ist rechts neben jeder Regel der Quelltext angegeben, der zum Erzeugen dieser Regel verwendet wurde.

- Die Numerierung von Regeln erfolgt zusammen mit den Formeln der `equation`-Umgebung. Das bedeutet, daß nach der Gleichung (3)

$$x = \sqrt[3]{a^3 + b^3 + c^3} \quad (3)$$

die Regel (4) folgt.

<pre>IF (Bedingung) THEN (Folgerung)</pre>	(4)	<pre>\begin{regel} \IF{Bedingung} \THEN{Folgerung} \label{regel13} \end{regel}</pre>
---	-----	--

- Die Numerierung einer Regel kann durch das Kommando `\nonumber` in der normalen `regel`-Umgebung bzw. durch die Verwendung der `regel*`-Umgebung unterdrückt werden. Die beiden folgenden Regeln bekommen beispielsweise keine Nummer:

<pre>IF (Bedingung) THEN (Folgerung)</pre>	<pre>\begin{regel} \IF{Bedingung} \THEN{Folgerung} \nonumber \end{regel}</pre>
<pre>IF (Bedingung) THEN (Folgerung)</pre>	<pre>\begin{regel*} \IF{Bedingung} \THEN{Folgerung} \end{regel*}</pre>

- In einer `regel`-Umgebung können mehrere Regeln hintereinander angegeben werden, die dann gemeinsam eine Regelnummer erhalten. Das bedeutet, nicht Regeln sondern `regel`-Umgebungen werden nummeriert.
- *Das in einer `regel`-Umgebung mögliche `\label`-Kommando muß direkt vor dem `\end{regel}`-Befehl angegeben werden (siehe Beispiele).*
- Um Regeln in den Text mit einbeziehen zu können, ist nach dem Argument eines `\IF`- bzw. `\THEN`-Kommandos ein optionales Satzzeichen aus `{.,;}` erlaubt. Dieses Satzzeichen erscheint im Regel-Text nach der abschließenden rechten Klammer des entsprechenden Arguments. Beispiele dafür bieten die Regeln (5) bis (7).

<pre>IF (Bedingung A), THEN (Folgerung B).</pre>	(5)	<pre>\begin{regel} \IF{Bedingung A}, \THEN{Folgerung B}. \label{regel14} \end{regel}</pre>
---	-----	--

IF (Bedingung C). THEN (Folgerung D),	(6)	<pre> \begin{regel} \IF{Bedingung C}. \THEN{Folgerung D}, \label{regel5} \end{regel} </pre>
IF (Bedingung E); THEN (Folgerung F);	(7)	<pre> \begin{regel} \IF{Bedingung E}; \THEN{Folgerung F}; \label{regel5} \end{regel} </pre>

- Die Platzierung von Regeln im Text kann durch folgende Maße beeinflusst werden (ihre Defaultwerte sind jeweils in Klammern dahinter angegeben):
 - `\aboveruleskip` (1.5ex) und `\belowruleskip` (1.5ex)
Abstand einer als eigener Absatz gesetzten Regel zum vorhergehenden bzw. nachfolgenden Absatz.
 - `\aboveruleshortskip` (0.5ex) und `\belowruleshortskip` (0.5ex)
Abstand einer zum vorhergehenden bzw. nachfolgenden Absatz dazugehörigen Regel zu diesem Absatz.
 - `\ruleleftindent` (1cm) und `\rulerrightindent` (2cm)
Breite der linken und rechten Einrückung einer Regel bezüglich des aktuellen linken und rechten Textrandes. `\rulerrightindent` sollte nicht zu klein gewählt werden, da im rechten Rand die Regelnummer erscheint.
 - `\inruleskip` (0pt)
Zusätzlicher Abstand zum normalen Zeilenabstand zwischen dem `\IF`- und `\THEN`-Teil einer Regel.
 - `\betweenruleskip` (1ex)
Abstand zwischen zwei Regeln innerhalb der gleichen `regel`-Umgebung.
 - `\rulehskip` (1ex)
Abstand zwischen den Schlüsselwörtern THEN, AND, OR und dem nachfolgenden Text.

Diese Maße können über das Kommando
`\setlength{Mass}{Laenge}`
 verändert werden.

Als abschließende Beispiele noch eine größere Regel, in der der Text von Bedingungen und Folgerungen umbrochen werden muß, sowie einige Regeln, die direkt aneinanderstoßen.

IF (eine sehr lange Bedingung, die eigentlich schön umbrochen werden müßte
AND eine zweite sehr lange Bedingung, die auch schön umbrochen werden müßte),
THEN (eine sehr lange Folgerung, die wiederum schön umbrochen werden müßte
OR eine zweite sehr lange Folgerung, die schließlich auch schön umbrochen werden müßte
OR eine dritte lange Folgerung mit der Regelnummer daneben). (8)

IF (Bedingung A);
THEN (Folgerung B); (9)

IF (Bedingung D);
THEN (Folgerung C); (10)

IF (Bedingung E);
THEN (Folgerung F); (11)

Struktogramme setzen mit `strukto.sty`

Thomas Birnthaler

Für: \LaTeX

Aufgabe: Stellt eine Umgebung zum Setzen von Struktogrammen zur Verfügung.

Struktogramme dienen zur übersichtlichen Darstellung von Programmen in graphischer Form. Aus einigen wenigen einfachen graphischen Grundelementen lassen sich damit durch Aneinanderreihung und Verschachtelung beliebig komplexe Programmstrukturen beschreiben. Das folgende einfache Struktogramm

READ untergrenze
READ obergrenze
FOR i = untergrenze TO obergrenze
IF Odd(i)
THEN print 3 * i + 1
ELSE print i / 2

wurde mit der `strukto`-Umgebung der hier beschriebenen Dokumentstil-Option durch folgende Anweisungen erstellt:

```

\begin{struktogramm}{6cm}{0.5cm}
  \BLOCK {READ untergrenze\
          READ obergrenze}
  \FOR {i = untergrenze TO obergrenze}
  DO {\IF {Odd(i)}
      THEN {\BLOCK* {print 3 * i + 1}}
      ELSE {\BLOCK* {print i / 2}}}
\end{struktogramm}

```

In den folgenden Beispielen ist rechts neben jedem Struktogramm der zugehörige Quelltext angegeben. Zur Erzeugung von Struktogrammen werden von der `struktogramm`-Umgebung folgende graphische Grundelemente zur Verfügung gestellt:

elementare Anweisung 1 ... elementare Anweisung N	<code>\BLOCK {elementare Anweisung 1 \\ \ldots \\ elementare Anweisung N}</code>
FOR Bedingung Anweisung	<code>\FOR {Bedingung} DO {\BLOCK {Anweisung}}</code>
WHILE Bedingung Anweisung	<code>\WHILE {Bedingung} DO {\BLOCK {Anweisung}}</code>
Anweisung UNTIL Bedingung	<code>\REPEAT {\BLOCK {Anweisung}} UNTIL {Bedingung}</code>
LOOP Anweisung	<code>\LOOP {\BLOCK {Anweisung}}</code>
IF Bedingung THEN Anweisung ELSE Anweisung	<code>\IF {Bedingung} THEN {\BLOCK* {Anweisung}} ELSE {\BLOCK* {Anweisung}}</code>

Diese Konstrukte können zur Erzeugung eines beliebigen Struktogramms entsprechend den folgenden Regeln aneinandergereiht und ineinandergeschachtelt werden:

- Die Befehle zur Erzeugung eines Struktogramms stehen *nur* in der `struktogramm`-Umgebung zur Verfügung. Beim Einleitungsbefehl der Umgebung *müssen* die gewünschte Breite dieses Struktogramms und die gewünschte Einrückungsbreite von Schleifenkörpern angegeben werden. Im folgenden Beispiel beträgt die Struktogrammbreite 6 cm und die Einrückungsbreite 0.5 cm:

```
\begin{struktogramm}{6cm}{0.5cm}
...
\end{struktogramm}
```

In den Quelltexten der Beispiele wurden diese beiden Befehle zur Einleitung und Beendigung der `struktogramm`-Umgebung der Übersichtlichkeit halber weggelassen.

- Eine komplette Struktogrammumgebung kann als eine Einheit betrachtet werden, die wie ein großer Einzelbuchstabe in den umgebenden Text eingefügt wird (Basislinie ist der obere Rand). Da dies wenig sinnvoll erscheint, wird sie am besten in eine `figure`-Umgebung eingeschachtelt und mit einem `\centering`-Befehl zentriert zum umgebenden Text dargestellt:

```
\begin{figure}[htbp]
\centering
\begin{struktogramm}{6cm}{0.5cm}
...
\end{struktogramm}
\end{figure}
```

- Elementare Anweisungen *müssen* in ein `\BLOCK`-Konstrukt eingeschachtelt werden. Wird dies vergessen, kann der Programmtext nicht korrekt im Struktogramm plaziert werden. Ein Beispiel dafür, wie man es also nicht machen sollte, ist

FOR $i = 1$ TO 10
Print i
Print i^2

```
\FOR {i = 1 TO 10}
DO {Print $i$ \
  Print $i^2$}
```

- Eine Sequenz von aufeinanderfolgenden Konstrukten wird (ohne Leerzeile dazwischen) einfach untereinandergeschrieben.

$A = 1, B = 2$
$C = 1$
IF $A \leq B \wedge B \leq C$
THEN Print "sorted"
ELSE Print "muddled"

```
\BLOCK {$A = 1, B = 2$}
\BLOCK {$C = 1$}
\IF {$A \le B \wedge B \le C$}
THEN {\BLOCK* {Print 'sorted'}}
ELSE {\BLOCK* {Print 'muddled'}}
```

- Als *Bedingung* in IF-, FOR-, WHILE- und REPEAT-Konstrukten ist beliebiger Text erlaubt, aber (sinnvollerweise) keine Struktogrammanweisung. Siehe die FOR- und IF-Anweisung in den beiden vorhergehenden Beispielen.
- In den Anweisungsteilen der IF-, FOR-, WHILE-, REPEAT- und LOOP-Konstrukte, sowie in den THEN- und ELSE-Zweigen des IF-Konstrukts

sind wiederum beliebige Struktogramm-Konstrukte erlaubt. Im einfachsten Fall ist dies eine `\BLOCK`-Kommando mit Programmtext. Ein Beispiel für eine komplexere Verschachtelung von Konstrukten ist

Read i
WHILE $i > 1$
IF Odd(i)
THEN $i = 3 * i + 1$
ELSE $i = i / 2$
UNTIL $i = 0$

```
\REPEAT{%
  \BLOCK {Read $i$}
  \WHILE {$i > 1$}
  DO {\IF {Odd($i$)}
    THEN {\BLOCK* {$i = 3 * i + 1$}}
    ELSE {\BLOCK* {$i = i / 2$}}}}
UNTIL{$i = 0$}
```

Nebenbei: Es ist nicht bekannt, ob nach Eingabe eines beliebigen Wertes für i jemals ein zweiter Wert eingegeben werden kann.

- Wird nur der THEN-Teil eines IF-Konstruktes benötigt, *muß* ein ELSE-Teil mit leerem Argument angegeben werden. Der ELSE-Teil wird dann im Struktogramm weggelassen.

IF Odd(i)
THEN $i = 3 * i + 1$

```
\IF {Odd($i$)}
THEN {\BLOCK* {$i = 3 * i + 1$}}
ELSE {}
```

- Zum IF-Konstrukt gibt es eine *extended*-Form, bei der links vom gesamten IF-Konstrukt eine schmale Spalte erscheint (halb so breit wie die Einrückungsbreite von Schleifenkörpern).

IF $B < 0$
THEN $B = B + 1$
ELSE $B = B - 1$

```
\IFx {$B < 0$}
THEN {\BLOCK* {$B = B + 1$}}
ELSE {\BLOCK* {$B = B - 1$}}
```

- Bei beiden Formen des IF-Konstrukts kann optional ein `*` nach dem Schlüsselwort IF/IFx angegeben werden, der dafür sorgt, daß die IF-, THEN- und ELSE-Teile alle gleich breit werden und zwischen dem IF-Schlüsselwort und der Bedingung ein Trennstrich gezogen wird.

IF	$B < 0$
THEN	$B = B + 1$
ELSE	$B = B - 1$
IF	$B < 0$
THEN	$B = B + 1$
ELSE	$B = B - 1$

```
\IF* {$B < 0$}
THEN {\BLOCK {$B = B + 1$}}
ELSE {\BLOCK {$B = B - 1$}}
\IFx* {$B < 0$}
THEN {\BLOCK {$B = B + 1$}}
ELSE {\BLOCK {$B = B - 1$}}
```

- Zum BLOCK-Konstrukt gibt es eine *-Form, die dafür sorgt, daß links vom Block *kein* Strich erscheint. Dieser `\BLOCK*`-Befehl ist *nur* im normalen IF-Konstrukten sinnvoll anwendbar. Siehe die IF-Konstrukte in vorhergehenden Beispielen.
- Zu jedem der Befehle `\FOR`, `\WHILE`, `\REPEAT` und `\LOOP` kann optional ein (kurzer) Text angegeben werden, der das jeweilige Standard-Schlüsselwort `FOR`, `WHILE`, `UNTIL` oder `LOOP` ersetzt. Dieser optionale Parameter darf auch leer sein.

\forall Bedingung	
	Anweisung
SOLANGE Bedingung	
	Anweisung
	Anweisung
BIS Bedingung	
WIEDERHOLE	
	Anweisung

```

\FOR[$forall$] {Bedingung}
DO {\BLOCK {Anweisung}}
\WHILE[SOLANGE] {Bedingung}
DO {\BLOCK {Anweisung}}
\REPEAT[BIS] {\BLOCK {Anweisung}}
UNTIL {Bedingung}
\LOOP[WIEDERHOLE]
{\BLOCK {Anweisung}}
    
```

- Zu den Befehlen `\IF` bzw. `\IFx` können optional drei durch Komma getrennte (kurze) Texte angegeben werden, die die Standard-Schlüsselworte `IF`, `THEN` und `ELSE` ersetzen.

WENN $B < 0$	
DANN $B = B + 1$	
SOLANGE $B = B - 1$	
???	$B < 0$
\Rightarrow	$B = B + 1$
\nRightarrow	$B = B - 1$

```

\IF[WENN,DANN,SOLANGE] {$B < 0$}
THEN {\BLOCK* {$B = B + 1$}}
ELSE {\BLOCK* {$B = B - 1$}}
\IFx[{\bf ???},
      $\Rightarrow$,
      $\not\Rightarrow$] {$B < 0$}
THEN {\BLOCK* {$B = B + 1$}}
ELSE {\BLOCK* {$B = B - 1$}}
    
```

- Das Maß `\struktohskip` (0.5ex) bestimmt den (Mindest)Abstand von Schlüsselworten und Programmtexten vom linken und rechten Rand des umgebenden Kastens. Es kann mit dem Kommando `\setlength{\struktohskip}{Laenge}` innerhalb der `strukto`-Umgebung beliebig verändert werden.

IF	$B < 0$
THEN	$B = B + 1$
ELSE	$B = B - 1$

```

\struktohskip=1em
\IF* {$B < 0$}
THEN {\BLOCK {$B = B + 1$}}
ELSE {\BLOCK {$B = B - 1$}}
    
```

Zum Abschluss noch ein große Struktogramm, das den berühmten Backtracking-Algorithmus zur Suche eines Pfades in einem Graphen darstellt:

Initialisiere: ENDE := 'false', FAILURE := 'true'			
Ermittle die Liste NACHFOLGER der direkten Nachfolger vom übergebenen Knoten K und ordne sie entsprechend der Funktion BEWERTUNG			
WHILE ENDE = 'false' und NACHFOLGER nicht leer			
N sei der erste Knoten aus NACHFOLGER			
IF	N ist ein terminaler Knoten		
THEN	ENDE := 'true' FAILURE := 'false'		
ELSE	IF	Rückziehbedingung ist für N erfüllt	
	THEN	ENDE := FAILURE := 'true'	
	ELSE	IF	RZ(N) = 'false'
		THEN	Streiche N aus NACHFOLGER
ELSE	Erfolgreichen Pfad gefunden ENDE := 'true', FAILURE := 'false'		
Liefere FAILURE als Ergebnis zurück			

Figure 1: Ein allgemeiner Rückziehungs-Algorithmus $RZ(K)$ als rekursive boolesche Funktion mit übergebenem Knoten K

Bearbeitung mehrteiliger \LaTeX -Dokumente mit \TeXPART.BAT

Thomas Birnthaler

Für: \LaTeX unter MS-DOS.

Aufgabe: Mehrteilige \LaTeX -Dokumente effizient übersetzen.

Unter \LaTeX werden sehr lange Dokumente in der Regel in einzelne Teile zerlegt, die in getrennten Dateien (z.B. `part1.tex`, `part2.tex`, usw.) abgespeichert werden. Eine Hauptdatei (z.B. `main.tex`) faßt dann über `\include`-Kommandos alle Teildateien zur Übersetzung zusammen.

Da während der Erstellung eines Dokuments zur gleichen Zeit nur ein Textteil bearbeitet wird, muß auch nur dieser übersetzt werden, um ihn ausdrucken und korrigieren zu können. Dazu wird man in der Hauptdatei `main.tex` mit einem `\includeonly`-Kommando diesen Textteil auswählen und die Hauptdatei anschließend übersetzen.

Damit die Hauptdatei deswegen nicht immer wieder editiert werden muß, kann das `\includeonly`-Kommando auch aus einer anderen Datei über `\input` eingelesen werden. Diese Datei enthält dann das `\includeonly`-Kommando mit dem gewünschten Teiltex als Argument.

Die Batch-Datei `TEXPART.BAT` ermöglicht genau diese Vorgehensweise. Sie übersetzt normalerweise die Hauptdatei `main.tex`. Soll ein anderer Name für die Hauptdatei verwendet werden, kann er der Umgebungsvariablen `TEXMAIN` zugewiesen werden (`SET TEXMAIN=...`). Ebenso können die Namen der Literatur- und Index-Textteile über die Umgebungsvariablen `TEXTLIT` und `TEXINDEX` gewählt werden, wenn ihre Defaultnamen `lit.tex` und `index.tex` nicht verwendet werden sollen. Die Aufrufsyntax von `TEXPART` lautet:

```
TEXPART datei1 [datei2 ...]
TEXPART some datei1 [datei2 ...]
TEXPART not datei1 [datei2 ...]
TEXPART all
TEXPART bib
TEXPART idx
```

Als Argumente können beim Aufruf von `TEXPART` angegeben werden:

- Die Namen eines oder mehrerer zu übersetzender Text-Teile *ohne die Extension* `.tex` (z.B. `part1 part2 ...`). `TEXPART` erzeugt für jeden einzelnen dieser Text-Teile eine Datei `includes.tex` mit dem entsprechenden `\includeonly`-Kommando und übersetzt dann `main.tex`. Die dabei entstehenden Dateien `main.dvi/.log` werden anschließend (wenn kein Fehler beim Übersetzen aufgetreten ist) umbenannt in `part1.dvi/.log`, usw.
- **SOME** führt zur Übersetzung der danach angegebenen Text-Teile in einem Stück und benennt die entstehenden Dateien `main.dvi/.log` in `some.dvi/.log` um.
- **NOT** führt zur Übersetzung aller Text-Teile *außer* den danach angegebenen Text-Teilen in einem Stück und benennt die entstehenden Dateien `main.dvi/.log` in `not.dvi/.log` um.
- **ALL** führt zur Übersetzung des gesamten Dokuments in einem Stück und benennt die entstehenden Dateien `main.dvi/.log` in `all.dvi/.log` um.
- **BIB** dient zum Erstellen des Literaturverzeichnisses aus den im Text angegebenen `\cite`-Befehlen. Es führt zum Aufruf des Programms `BIBTEX` vor der Übersetzung des Textteils `lit.tex`.
- **IDX** dient zum Erstellen des Indexes aus den im Text angegebenen `\index`-Befehlen. Es führt zum Aufruf des Programms `MAKEIND(E)X` vor der Übersetzung des Textteils `index.tex`.

Auf der beiliegenden Diskette findet man neben der Batch-Datei `TEXPART.BAT` die Haupt-Textdatei `main.tex` mit ihren Hilfs-Textdateien `sizes.own`, `commands.own`, `hyphen.own`, sowie `titel.tex`, `toc.tex`, `lit.tex` und `index.tex`. Die T_EX-Dateien sind als Musterbeispiele zu betrachten und an die eigenen Gegebenheiten anzupassen.

Das ebenfalls beigefügte Programm MAKEIDX.EXE und die Dateien `makeidx.sty` und `inclndot.sty` sind zum Erstellen eines Indexes bzw. für den Parameter **NOT** notwendig.

Hinweis: `TEXPART` verwendet zur Messung der zum Übersetzen benötigten Zeit das Programm `TM` aus den Norton-Utilities.

EXPDLIST – eine Erweiterung der `description`-Umgebung

Rainer Hülse und Wolfgang Kaspar

Zusammenfassung

Die erweiterte `description`-Umgebung soll die `LATEX`-`description`-Umgebung nicht ersetzen, sondern bietet bei Bedarf einige zusätzliche Möglichkeiten. Daneben steht mit `\listpart` ein neuer, für alle `list`-Umgebungen gültiger Befehl zur Verfügung.

Der benötigte `STY`-File heißt `EXPDLIST` und wird so in den `LATEX`-File eingebunden:

```
\documentstyle[...expdlist,...]{dokumentstil}
```

Die Syntax der erweiterten `description`-Umgebung ist:

```
\begin{description}[deklarationen]
:
\end{description}
```

Ohne die optionalen `[deklarationen]` verhält sich diese Umgebung wie die originale `LATEX` `description`-Umgebung. Folgende Deklarationen sind möglich:

`\compact` zeigt an, daß die Definitionen nicht durch Leerzeilen voneinander getrennt werden. Definitionen werden normalerweise durch Leerzeilen voneinander getrennt.

`\setlabelsize{länge}` gibt die Länge des horizontalen Freiraums an, der für `\labelwidth` und `\labelsep` freigehalten wird.¹ Die Voreinstellung entspricht der originalen `description`-Liste in `LATEX`.

`\setlabelphantom{text}` reserviert die Länge von `text` als horizontalen Freiraum für die Marke ohne `\labelsep`.¹ Schriftstil und -größe von `text` richten sich nach der Einstellung von `\setlabelstyle`.

¹ Wenn sowohl `\setlabelphantom` als auch `\setlabelsize` gesetzt sind, wird ein Freiraum der Länge, die durch `\setlabelphantom` definiert ist, freigehalten.

`\setlabelstyle{schriftstil}`

ist der Stil, der für die Marken benutzt wird, z. B. `\bf`, `\it`, `\sl`,
oder `\sf`, sowie `\small`, `\large` usw. Voreingestellt ist `\bf` und
`\normalsize`.

`\breaklabel`

läßt die Beschreibung in der nächsten Zeile beginnen, wenn
die Länge der Marke den Wert überschreitet, der durch
`\setlabelsize` festgelegt wurde. In der Voreinstellung wird ein
Leerzeichen hinter der Marke freigelassen. Die Beschreibung be-
ginnt dann auf der gleichen Zeile.

Die folgenden Beispiele zeigen einige Anwendungen der erweiterten
`description`-Umgebung.

Das erste Beispiel zeigt, daß sie ohne optionalen Parametern der originalen
L^AT_EX-Umgebung entspricht. Die abgesetzte Markierung lautet:

```
\begin{description}
```

Erste Marke Die erste Marke ist durchschnittlich lang.

Hier nun eine besonders lange Marke Dies ist der Text, der zu der
besonders langen Marke gehört.

3. Die 3. Marke ist sehr kurz.

Dieser Eintrag hat keine Marke und wurde erzeugt mit `\item text`.

Im zweiten Beispiel werden mit der folgenden Markierung optionale Parameter
gesetzt:

```
\begin{description}
```

```
[\breaklabel\setlabelsize{80pt}\setlabelstyle{\it}]
```

Erste Marke Die erste Marke ist durchschnittlich lang.

Hier nun eine besonders lange Marke

Dies ist der Text, der zu der besonders langen Marke
gehört.

3. Die 3. Marke ist sehr kurz.

Dieser Eintrag hat keine Marke und wurde erzeugt
mit `\item text`.

Das letzte Beispiel zeigt die Markierung mit weiteren optionalen Parametern
und ihre Wirkung:

```
\begin{description}[\compact\setlabelphantom{Erste Marke}]
```

Erste Marke Die erste Marke ist durchschnittlich lang.

Hier nun eine besonders lange Marke Dies ist der Text, der zu der besonders langen Marke gehört.

3. Die 3. Marke ist sehr kurz.

Dieser Eintrag hat keine Marke und wurde erzeugt mit `\item text`.

Das `\listpart`-Kommando

Der EXPDLIST-Style enthält noch zwei weitere neue L^AT_EX-Kommandos:

`\listpart{text}` ist ein Kommentar oder eine Erklärung, die als Teil einer Liste gilt. Er kann irgendwo in einer beliebigen `list`-Umgebung stehen, direkt hinter dem Listeneintrag, zu dem er gehört. Die Zeilenbreite von `text` richtet sich dabei nach der Breite der übergeordneten Liste. Man kann somit überleitenden Text zwischen zwei Listeneinträgen setzen, ohne die Liste beenden und anschließend wieder neu beginnen zu müssen. Die Numerierung in der `enumerate`-Umgebung bleibt dabei erhalten.

`\listpartsep` ist der vertikale Abstand zwischen Listeneintrag und dem mit `\listpart` erzeugten Kommentar. Voreingestellt ist `1ex`.

Das folgende Beispiel zeigt, daß man `\listpart` auch in geschachtelten `list`-Umgebungen benutzen kann:

- Man kann `\listpart` in einer geschachtelten Liste benutzen.

Dies ist ein `listpart`. Mit dieser Markierung wird Text eingeschoben, der die Liste unterbricht.

- Man kann `\listpart` benutzen in:

1. `itemize`-Listen
2. `enumerate`-Listen

Dies ist ein `listpart`. Mit dieser Markierung wird Text eingeschoben, der die Liste unterbricht.

3. `description`-Listen:
 1. **Marke** Beschreibung A

Dies ist ein `listpart`. Mit dieser Markierung wird Text eingeschoben, der die Liste unterbricht.

2. **Marke** Beschreibung B

Anm. der Redaktion: Die beschriebenen Styles und die `.BAT`-Prozedur sind auf dem Listserver in Heidelberg verfügbar.

Rezensionen

\TeX -Software im Vergleich

x \TeX , y \TeX , z \TeX , ...

Wieviele weitere Versionen wird es noch geben?

Joachim Lammarsch

Nachdem es immer mehr Anfragen nach den unterschiedlichen Versionen von \TeX unter dem Betriebssystem PC/MSDOS gab, habe ich mich entschlossen, die gängigsten Versionen zu installieren und zu vergleichen, welche wirklich die schnellste ist.

als Gerät benutzte ich einen IBM PS/2, Model 70 mit 20 MHz Taktfrequenz, ein relativ schnelles Gerät mit schneller Festplatte. Außerdem war genügend Hauptspeicher für eine RAM-Disk vorhanden. Alle Versionen installierte ich auf dem Laufwerk *d*.

Die Installation (auf Festplatte und Erzeugen der FMT-Files für plain und \LaTeX mit englischer Silbentrennung):

1. Kommerzielle Versionen:

- C \TeX 1.8 (entspricht \TeX 2.8), vertrieben von der Firma Breakpoint, Holzkirchen
War schnell installiert, beim Erzeugen der FMT-Files ergaben sich keine Probleme.
- P \TeX 2.93, vertrieben von der Firma Breakpoint, Holzkirchen
War schnell installiert, beim Erzeugen der FMT-Files ergaben sich keine Probleme.
- Turbo \TeX 2.0, vertrieben von Kinch Inc., Ithaca, N.Y.
Hier war eine Installation auf dem *d*-Laufwerk nicht vorgesehen. Es half nur eine Änderung der Installationsprozedur. Die Installation benötigte im Vergleich zu den beiden vorigen Versionen viel Zeit. Beim Erzeugen der FMT-Files ergaben sich keine Probleme.
- $\mu\TeX$, vertrieben von der Firma MID/Information Logistics Group GmbH, Heidelberg
Hier dauerte die Installation am längsten. Allerdings werden bei dieser Version die Fonts für den integrierten Bildschirmtreiber gleich mitinstalliert. Für das Erzeugen der FMT-Files sind zwei Kommandoprozeduren (BAT-Files) vorhanden, die aber bei mir nicht funktionierten. Abhilfe schuf der Verzicht auf die BAT-Prozeduren und Aufruf von Hand.

2. *public domain* Versionen:

- $\text{DOST}_{\text{E}}\text{X}$
Da mir davon nur eine alte Version zur Verfügung stand, die sich für den Vergleich nicht eignet, werde ich die Daten in einer der nächsten Ausgaben der Zeitung nachreichen.
- $\text{EMT}_{\text{E}}\text{X}$, von Eberhard Mattes
Der neueste Stern am Himmel der $\text{T}_{\text{E}}\text{X}$ -Versionen. Nach Aussagen des Autors die schnellste, beste und vollständigste Version. Die Installation ging schnell. Die FMT-Files sind schon vorhanden.
- $\text{PubliCT}_{\text{E}}\text{X}$, von Klaus Thull
Die Version, die von DANTE e.V. verteilt wird. Die Installation ging schnell, ebenso das Erstellen der FMT-Files.
- $\text{SBT}_{\text{E}}\text{X}$, von Wayne Sullivan
Die Installation ging schnell. Allerdings fehlten eine ganze Menge Files, die ich mir von $\text{PubliCT}_{\text{E}}\text{X}$ ausgelohnt habe. Beim Erzeugen des FMT-Files ergaben sich keine Probleme.

Zum Zeitvergleich hatte ich mir drei weit verbreitete Eingabetexte ausgesucht:

- *The T_EXbook*, D. E. Knuth (A)
- A Gentle Introduction to $\text{T}_{\text{E}}\text{X}$, Michael Doob (B)
- Kurzfürher $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, Hubert Partl (C)

Diese Eingaben haben den Vorteil, daß sie nicht zu kurz sind und gewisse Anforderungen an die verwendete $\text{T}_{\text{E}}\text{X}$ -Version stellen. Zwei der kommerziellen Versionen – $\text{CT}_{\text{E}}\text{X}$ und Turbo $\text{T}_{\text{E}}\text{X}$ – scheiterten an *The T_EXbook*. Sie waren nicht in der Lage, es ohne Fehler zu bearbeiten. Aus diesem Grund habe ich sie zurückgestellt, bis die Fehler mit den entsprechenden Firmen abgeklärt sind.

Alle Versionen sind in der Lage, einen Teil ihres Speichers auf der Festplatte abzulegen. $\text{PCT}_{\text{E}}\text{X}$ und $\text{EMT}_{\text{E}}\text{X}$ unterstützen *expanded memory* nach der EMS-Norm. Gemessen wurde die Zeit für „paging“ auf die Festplatte (1. Wert), auf eine virtuelle Disk (2. Wert) und wenn möglich in *expanded memory* (3. Wert). Daraus ergaben sich folgende Zeiten (in min:sec):

	PCT _{TEX}			μ TEX		
A	18:05	17:56	17:57	105:59	40:04	–
B	1:46	1:45	1:45	3:30	3:29	–
C	2:59	2:54	2:59	48:01	10:15	–

	EMTEX			PubliCT _{TEX}			SBTEX		
A	15:09	15:11	15:09	26:38	26:36	–	19:05	15:12	–
B	1:40	1:41	1:41	2:43	2:43	–	1:22	1:20	–
C	2:28	2:28	2:28	4:31	4:23	–	3:26	2:48	–

Fazit:

Am schnellsten sind SBTEX und EMT_{TEX}, gefolgt von PCT_{TEX}. PubliCT_{TEX} liegt im Mittelfeld, dann kommt mit Abstand μ TEX. CT_{TEX} und Turbo TEX, genauso wie μ TEX müssen nochmals überprüft werden. Auch werde ich alle Tests auf einem langsameren Rechner wiederholen, in der Hoffnung, daß man die Vorteile einer virtuellen Disk bzw. des *expanded memory* sehen kann.

Fazit:

Trotz der Ergebnisse wird auch weiterhin PubliCT_{TEX} von DANTE e.V. verteilt werden.¹ Ein wichtiger Grund dafür ist, daß nur bei dieser *public domain* Version der Quellcode vorhanden ist und weitergegeben werden kann. Bei einer Verteilung im Sinne von D. E. Knuth erwarte ich die Möglichkeit, mir die Eingabe meines TEX-Systems anzusehen, um bei Bedarf die Parameter ändern zu können. Dies ist sowohl bei SBTEX als auch bei EMT_{TEX} nicht möglich.

Bei EMT_{TEX} kommt noch hinzu, daß das ganze Paket nur geschlossen weitergegeben werden darf. Es entspricht aber sowohl die L^ATEX-Version als auch die Anpassung an die deutsche Sprache nicht dem im deutschsprachigen Raum gültigen Standard. Aufgrund des Copyright-Verwerks des Autors ist ein Austausch auch nicht möglich.

Ich persönlich finde nicht, daß all die verschiedenen Versionen ein besonderer Zuwachs für TEX sind. Im kommerziellen Bereich ist es noch verständlich, wenn mehrere Firmen ihr Stückchen vom Kuchen verdienen wollen. Im *public domain* Bereich schiene es mir erheblich sinnvoller, wenn alle ihre Ideen dazu beisteuern würden, *ein* gutes *public domain* TEX zu entwickeln. Aber wenn das jetzt Mode geworden ist, sollte ich mir wohl überlegen, auch ein eigenes j_{TEX} zu entwickeln!?

¹ SBTEX und EMT_{TEX} werden über DANTE e.V. erhältlich sein, allerdings ohne Unterstützung.

T_EX-Beiprogramm

Publisher

MID/Information Logistics Group GmbH

Der *Publisher* von ArborText ist ein hochentwickeltes, T_EX-basiertes Dokumentenverarbeitungssystem für wissenschaftliche und technische Publikationen. Dokumente mit Formeln, Tabellen und Graphiken werden WYSIWYG erstellt und bearbeitet. Darüberhinaus kann *Publisher* in T_EX, L_AT_EX oder BibT_EX ausgezeichnete Texte übernehmen.

Bei der Erstellung eines Dokuments kann beliebig zwischen WYSIWYG- und T_EX-Eingabe gewechselt werden. So können innerhalb eines Textes einige Passagen nach Art komfortabler Textverarbeitungsprogramme und andere Passagen in T_EX geschrieben werden. Oder Publikationen werden von verschiedenen Autoren erstellt: T_EX-Spezialisten schreiben in T_EX, andere Autoren wissen die Vorteile der WYSIWYG-Oberfläche mit den verschiedenen Spezialeditoren z.B. für Formeln, Tabellen und Graphik zu schätzen. Am Ende wird im *Publisher* alles zu einer einheitlich gestalteten Publikation zusammengeführt.

Damit die einzelnen Publikationen einer Organisation tatsächlich ihre jeweils eigene inhaltliche Struktur und formale Gestaltung wahren, können mit *Publisher* sogenannte Styles vorgegeben werden. Struktur und Formatierung der Publikationen werden vom Benutzer definiert und für die verschiedenen Dokumenttypen abgespeichert. Für die Arbeit mit *Publisher* ergeben sich klare Vorteile:

1. *Publisher* zeigt im Editor die Struktur des zu erstellenden Dokuments und unterstützt damit die Autoren bei ihrer Arbeit.
2. Dokumente desselben Typs werden einheitlich formatiert und gestaltet. Ein einheitliches Erscheinungsbild z.B. aller Publikationen einer Organisation ist somit automatisch gewährleistet.

Publisher unterstützt insbesondere die interaktive Erstellung und Bearbeitung von Formeln, Tabellen und Graphiken. Hier arbeitet *Publisher* mit spezialisierten Subeditoren, die aus dem Text heraus aufgerufen werden. So werden z.B. die Zeichen in mathematischen Formeln automatisch entsprechend den mathematischen Gepflogenheiten bezüglich Positionierung und Zeichengröße gesetzt.

Auch hier besteht zusätzlich die Möglichkeit, mathematische Formeln wie gewohnt direkt in T_EX-Kodierung einzugeben. Mit *Publisher* können jedoch auch T_EX-unerfahrene Autoren die Überlegenheit des Satzsystems T_EX gerade

in Hinblick auf den Satz von Formeln und Tabellen nutzen. Tatsächlich können selbst Schreibkräfte die mathematischen Symbole von einer Palette auswählen, ohne die Bedeutung der Symbole oder ihre Bezeichnungen in $\text{T}_{\text{E}}\text{X}$ zu kennen. Die Benutzerunterstützung und Menüführung kommt vom *Publisher*, im Hintergrund arbeitet $\text{T}_{\text{E}}\text{X}$.

Text und Graphik können im *Publisher* gemischt werden. PostScript, Encapsulated PostScript (EPS), Raster- und Vektorgraphiken können problemlos eingebunden werden.

1. *Publisher* ist vollständig SGML-basiert. SGML (*Standard Generalized Markup Language*, ISO-Standard 8879) ist eine systemunabhängige Sprache zur inhaltsbezogenen Auszeichnung von Dokumenten. *Publisher* kann SGML-Dokumente verarbeiten und seine eigenen Dokumente als SGML-Dateien ausgeben.
2. Durch Konvertierung können viele Textformate übernommen werden. Die wesentlichen Formatierungen aus den Textverarbeitungssystemen bleiben dabei erhalten.
3. Kataloge oder Verzeichnisse, die in Datenbanken gehalten werden, können mit *Publisher* direkt gesetzt und formatiert werden. Die Informationen aus der Datenbank müssen nicht neu erfaßt werden. Indices, Fußnoten, Inhaltsverzeichnis und Querverweise werden von *Publisher* automatisch verwaltet.

Die Dokumentausgabe erfolgt in $\text{T}_{\text{E}}\text{X}$, DVI, SGML oder PostScript. Eine Konvertierung nach $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$ ist möglich. Für Hewlett-Packard-Drucker sind spezielle Treiber erhältlich.

Die aktuelle Version *Publisher 3.0* ist für alle Sun-Workstations unter SunView und X11 und für die Hewlett-Packard-Workstations 9000 der Serie 300 (außer 310) unter X-Windows verfügbar. Die Portierung für Apollo-Workstations DN 2500, 3500 und 4500 ist für die erste Jahreshälfte 1990 angekündigt.

Die Einzellizenz des *Publisher* kostet rund 10.000 Mark. Für Universitäten bietet der deutsche Importeur (MID/Information Logistics Group GmbH, Ringstr. 19, 6900 Heidelberg, Tel. (06221) 166091, Fax (06221) 23921) umfangreichen Hochschulrabatt. Der Preis der Einzellizenz liegt hier bei DM 3.500.

Magazin

Großer Ideen-Wettbewerb

Er läuft noch: der Ideen-Wettbewerb für unser

LOGO



Für alle, die es vergessen haben oder es als DANTE-Neulinge noch nicht mitbekommen konnten: wir sind auf der Suche nach einem charakteristischen Schriftzug für:

DANTE

Deutschsprachige Anwendervereinigung T_EX e.V.,
 der sowohl für „DANTE“ allein, als auch für den ganzen Schriftzug „DANTE,
 Deutschspr...“ verwendbar ist. Dabei ist noch zu berücksichtigen, daß dieser
 Schriftzug auf alle möglichen Medien gebracht werden soll: Stempel, Briefpa-
 pier, usw.

Vorschläge bitte bald und reichlich an DANTE senden. Die Einsendefrist ist
 vorläufig ausgesetzt; in Anbetracht des stetig fortschreitenden Jahres wird viel-
 leicht auch der vorgesehene Preis (ein kalligraphischer Kalender) abgeändert.

Bücherecke

Buchbesprechung: Design for Desktop Publishing

Konrad Neuwirth

Auch dieses Mal geht es um ein englisches Buch, geschrieben von John Miles.
 Das Buch „Design for Desktop Publishing“ ist eine leicht verständliche, klare
 Einleitung in das Thema Design und Typographie und, wie der Titel sagt,
 spezialisiert auf die Bedürfnisse eines Benutzers von DTP Systemen.

Obwohl man T_EX ja nicht unbedingt zu solchen Programmen zählen darf,
 kann man das Buch jedem T_EX-Benutzer, und hier vor allem Neulingen auf
 dem Gebiet, empfehlen.

Es beschäftigt sich auf 102 Seiten mit so ziemlich allen wichtigen Grundla-
 gen, von den ersten Begriffen des Seitenlayouts über Anwendungen von Fonts
 bis hin zur Einbindung von Grafiken in Texte.

Das Buch ist nicht nur sehr praxisorientiert, sondern ist auch durch die
 Verwendung von vielen Beispielen sehr leicht zu verwenden. So finden sich z.B.
 im Abschnitt über die Vor- und Nachteile von asymmetrischem Layout einige
 Gegenüberstellungen von Seiten des gleichen Inhalts, nur mit verschiedenem

Layout. Neben den einzelnen Beispielen gibt es auch noch eine genaue Erklärung ihrer Eigenheiten. Auch aus Gegenüberstellungen von fehlerhaft und richtig gesetzten Texten kann man sehr viel lernen. Außerdem gibt es ein gutes Stichwortverzeichnis, wodurch man sich, vor allem als Anfänger, durch die Vielzahl der englischen Fachbegriffe manövrieren kann. Sogar eine Liste mit den Symbolen für richtige Fehlerkorrektur in Bürstenabzügen findet man im Buch.

Das Problem für L^AT_EX-Benutzer: daß es ihnen auf den ersten Blick kaum helfen wird. Durch die Lektüre kann man höchstens herausfinden, warum L^AT_EX die eine oder andere Sache so und nicht anders macht, sonst ist aber nicht allzu viel an Wissen herauszuholen.

Das liegt sicher daran, daß das Buch eben primär für Benutzer von WY-SIAYG (what you see is all you've got) gedacht ist. Das Buch ist auch auf einem MacIntosh mit einem solchen System entstanden (das sieht man an kleinen Details, z.B. welche Art von Anführungszeichen verwendet wird). Nur die letzten paar Seiten, die sich vor allem mit den Produktionsschritten nach dem Satz beschäftigen, bringen Neues. So lernt man einiges über die Vor- und Nachteile einzelner Vervielfältigungstechniken, aber auch über verschiedene Papiersorten.

Alles in allem ist das Buch „Design for Desktop Publishing“ eine leicht zu lesende Einführung in ein nicht so leichtes Gebiet. Es ist vor allem für Newcomer in der T_EX-Welt interessant, kann aber niemandem schaden (auch wenn man nachher nur über die Nachteile der WYSIWYG Systeme schimpfen will).

Design for Desktop Publishing

A guide to typography on the personal computer

John Miles

Chronical Books, San Francisco

ISBN: 0 87701-479-5

Preis 18.95 \$

T_EX-Poesie

Rotkäppchen wie der Informatiker es seinen Kindern erzählt

Es war einmal ein kleines süßes Mädchen, das immer ein Käppchen aus rotem Samt trug. Aufgrund dieses Attributes erhielt es ein Assign unter dem Namen „Rotkäppchen“.



Eines Tages sprach die Mutter: „Rotkäppchen, die Gesundheit Deiner Großmutter hat einen Interrupt bekommen. Wir müssen ein Pflege-Programm entwickeln und zur Großmutter bringen, um das Problem zu lösen. Verirre Dich jedoch nicht im Wald der alten Sprachen, sondern gehe nur strukturierte Wege. Nutze dabei immer eine Hochsprache der 4. Generation, dann geht es der Großmutter schnell wieder gut. Und achte darauf, daß Dein Pflege-Programm transaktionsorientiert ist, damit es die Großmutter nicht noch mehr belastet.“



Da der Weg zum Haus der Großmutter reentrent war, traf Rotkäppchen den bösen Wolf. Er tat sehr benutzerfreundlich, hatte im Background jedoch schon einen Abbruch programmiert. Während Rotkäppchen einen Go To ins Blumenfeld machte, ging der Wolf im Direktzugriff zur Großmutter und vereinnahmte sie unverzüglich durch einen Delete. Ohne zu zögern gab er sich den Anschein kompatibel zu sein und nahm die logische Sicht der Großmutter an. Dann legte er sich in ihren Speicherplatz.



Kurz danach lokalisierte auch Rotkäppchen die Adresse der Großmutter und trat in den Speicherraum. Vor Installation des Pflege-Programms machte Rotkäppchen sicherheitshalber einen Verify und fragte:

„Ei Großmutter, warum hast Du so große Augen?“—

„Weil ich zufriedene Endbenutzer gesehen habe.“

„Ei Großmutter, warum hast Du so große Ohren?“—

„Damit ich die Wünsche der User besser verstehen kann.“

„Ei Großmutter, warum hast Du so ein entsetzlich großes Maul?“—

„Damit ich Dich besser canceln kann!“

Sprach's und nahm das arme Ding als Input. Nach einem Logoff begab sich der Wolf zur Ruhe, schlief ein und begann laut zu schnarchen.



Als der Jäger auf seinem Loop durch den Wald am Haus der Großmutter vorbei kam, sah er durch ein Window den Wolf im Bett liegen. „Finde ich Dich hier, Du alter Sünder“, sprach er, „ich habe Dich lange gesucht!“ Als Kenner der Scene analysierte er sofort, daß nach den Regeln der Booleschen Algebra die Großmutter nur im Bauch des Wolfes sein konnte. Er nahm sein Messer, teilte den Bauch des Wolfes in mehrere Sektoren und machte, welch' Freude, die Großmutter und das Rotkäppchen wieder zu selbständigen Modulen.



Als Input für den leeren Bauch des Wolfes nahmen sie viele Kilobyte Steine und beendeten die Operation mit einem Close. Als der Wolf erwachte, verursachte ihm sein dermaßen aufgeblähter Hauptspeicher solche Schmerzen, daß er an einer Storage Violation jämmerlich zugrunde ging.



Da waren alle vergnügt. Das Pflege-Programm aktivierte die Großmutter. Rotkäppchen aber dachte: „Du willst dein Lebtag nie wieder ein Go To machen, sondern nur noch strukturierte Wege gehen, wie dir's die Mutter geboten hat.“

Der Fortsetzungsroman

Dante Aleghieri: Die göttliche Komödie

2. Folge

Und wie der Mensch noch mit gepreßtem Atem,
 Der sich vom Meere an den Strand gerettet,
 Zurückschaut nach den aufgeregten Wassern,
 So tat mein Geist, der, immer noch im Flüchten,
 Sich umgewandt, den Durchgang zu betrachten,
 Den nie ein Wesen lebend noch verlassen.
 Erst gab ich meinem müden Leibe Ruhe,
 Dann ging ich weiter durch die öde Wüste,
 Fest immer mit dem untern Fuße tretend.
 Und siehe, beinah beim Beginn des Hanges
 Erschien ein Panther, leicht und sehr behende,
 Der war bedeckt mit einem bunten Felle.
 Er wollt' mir nicht mehr aus den Augen gehen,
 Vielmehr hat er mir so im Weg gestanden,
 Daß ich mich mehrmals schon zur Umkehr wandte.
 Es war die Zeit der ersten Morgenstunde,
 Die Sonne stieg empor mit jenen Sternen,
 Die bei ihr waren, als die Liebe Gottes
 Die schöne Welt zum erstenmal bewegte.
 So hatt' ich Grund, auf Gutes nur zu hoffen
 Vom bunten Felle jenes wilden Tieres,
 Der Morgenstunde und dem schönen Frühling;
 Doch nicht so sehr, daß ich nicht fürchten mußte
 Das Bild des Löwen, der mir dann erschienen.
 Er schien gerade auf mich zuzukommen,
 Das Haupt erhoben und mit gierigem Hunger,
 Daß selbst die Luft vor ihm erbeben mußte.
 Und eine Wölfin kam, die schien beladen
 In ihrer Magerkeit mit allen Lüsten
 Und stürzte viele Menschen ins Verderben.

Fortsetzung folgt

Reclam-Verlag, 1987, Übersetzung von Hermann Gmelin, Anmerkungen von Rudolf Baehr. Der Nachdruck erfolgt mit freundlicher Genehmigung durch den Verlag Klett-Cotta, Stuttgart.

Spielplan

Tagungstermine

Korrekturen zum alten Spielplan sind *schräg* hervorgehoben.

- 15.-17.5.90 GUTenberg Tagung;
Toulouse, Frankreich;
Kontakt: Pierre Legrand.
- 18.-21.6.90 11. Jahrestreffen der T_EX Users Group;
Texas A & M University;
Kontakt: Christina Thiele.
- 31.8.90 NTG Tagung;
Groningen, Niederlande;
Kontakt: Kees van der Laan.
- 10.-13.9.90* TUG-Tagung „T_EX 90“;
Cork, Irland;
erste Tagung der TUG außerhalb Nord-Amerikas!
Kontakt: Joachim Lammarsch
- 10.-12.10.90* 9. Treffen deutschsprachiger T_EX-Interessenten
und 3. ordentliche Mitgliederversammlung von DANTE;
Gesellschaft für wissenschaftliche Datenverarbeitung, Göttingen.
Kontakt: Peter Scherber.
- 20.-22.2.91* 10. Treffen deutschsprachiger T_EX- Interessenten
und 4. ordentliche Mitgliederversammlung von DANTE;
Technische Universität Wien;
Kontakt: Hubert Partl.

Appel aux communications

Call for papers

GUTenberg'90

T_EX et les Arts graphiques

15 au 17 mai 1990

Université Paul Sabatier

TOULOUSE

The French-speaking T_EX Users Group named GUTenberg, is organizing its yearly congress comprising, as last year, tutorials and conferences, as well as a small exhibition.

This year, the congress will be entitled “T_EX and professional publishing” because T_EX has become a full set of software considered by printers, editors, composers... as an indispensable production tool. GUTenberg'90 will provide a unique opportunity to familiarize yourself with the products and to collect information about the latest experiments and developments.

The program committee will organize a series of high level presentations delivered by T_EX users or professionals of publishing.

Main topics

After giving priority to topics focused on T_EX and the publishing field, the program committee will select papers in one or the other of the following two areas:

Professional publishing

- Page modeling techniques
- Graphism and special effect
- New fonts usage
- Automated production from editor to printer
- Hardware experiences
- EP/CAP effects
- Markup standards

T_EX domain

- Developments around T_EX, L^AT_EX, ...
- Creation/usage of fonts designed with or without METAFONT
- Fonts and page description languages
- X-windows aspects
- Rasterization and output
- Screen editors, spellers and pre/post interactive processors
- Aspects of French and European languages
- Network usage
- Public domain software access

Organisation Committee

P. LEGRAND (*C.I.C.T.*), *président*
 Y. SOULET (*U.P.S.*)
 J.C. JOLY (*Cepadues éditions*)
 H. LE TALLEC (*Cepadues éditions*)
 M et MME COLLIN (*TRÉMA*)
 M.H. GELLIS
 P. LESGOURGUES (*Cepadues éditions*)

Contact

<i>Pierre Legrand</i>	<i>Bitnet:</i>
<i>C.I.C.T.</i>	LEGRAND@FRCICT81
<i>Univ. Paul Sabatier</i>	
<i>118, route de Narbonne</i>	<i>Tel: (33) 61.36.60.00</i>
<i>F31062 TOULOUSE Cedex</i>	<i>Fax: (33) 61.52.14.58</i>

Program Committee

BERNARD GAULLE (*CNRS-CIRCE*) *Chairman*
 MAURICE LAUGIER (*imprim. Louis-Jean*) *co-Chairman*
 JACQUES ANDRÉ (*IRISA-INRIA*)
 NELSON BEEBE (*Univ. Utha, USA*)
 ALAIN COUSQUER (*Univ. LILLE I*)
 NICOLAS BROUARD (*INED*)
 FRANÇOIS CHAHUNEAU (*Berger-Levrault*)
 MICHAEL FERGUSON (*INRS Tele., Canada*)
 LAURENT HEILMANN (*Gauthier-Villard*)
 CHRISTOPHE DE MONCUIT (*CNRS-LIMSI*)

Contacts

<i>Bernard Gaulle</i>	<i>Bitnet: UCIR001@FRORS31</i>
<i>CNRS-CIRCE, BP 167</i>	<i>Tel: (33-1) 69.82.41.07</i>
<i>F91403 ORSAY Cedex</i>	<i>Fax: (33-1) 61.28.52.73</i>
<i>Maurice Laugier</i>	
<i>Imprimerie Louis-Jean</i>	
<i>Avenue d'Embrun, BP 87</i>	<i>Tel: (33) 92.51.35.23</i>
<i>F05002 GAP</i>	<i>Fax: (33) 92.53.72.27</i>
<i>Nelson Beebe</i>	<i>Internet:</i>
<i>Cent. for Sc. Comput.</i>	Beebe@SCIENCE.UTAH.EDU
<i>220 South Physics Bldg</i>	
<i>Univ. of Utah</i>	<i>Tel: (1) 801 581-5254</i>
<i>Salt Lake City, UT 84112</i>	<i>Fax: (1) 801 581-4801</i>

Call for papers

A one page abstract must be sent to any committee member before **January 15, 1990** (please, indicate if the paper will be produced with \LaTeX).

Note the following deadlines: **January 15th:** abstract due
 February 1st: notification to authors
 February 28th: full paper due
 May 15–17th: GUTenberg'90

The official language is French but presentations in English are welcome.
 The association GUTenberg

The association GUTenberg is one of the main \TeX users groups. Due to the fact that GUTenberg is composed of French-speaking people it can work tightly with the *newly international* \TeX Users Group (TUG).

GUTenberg was officially created in 1988 and is a non-profit association of researchers, academic people and also professionals of publishing like editors, printers as well as companies working on various economic areas. One of the main goals of this association is to “*favour technical exchanges allowing the growth of printing facilities and of scientific publishing.*” In such a way, it is delighted to study all new products coming from the \TeX community and other software domains.

GUTenberg is the editor of a quarterly journal named *Cahiers GUTenberg*, distributor of various \TeX versions and other components of \TeX applications. It offers a discussion list on network which may be accessed from any network connected to EARN.

Advice for exhibitors

The show-room dedicated to exhibitors is relatively tiny. Please get in touch as quickly as possible with the organisation committee.

Sponsors

A lot of companies and public organizations intend to sponsor GUTenberg'90. The full list of these organizations will be printed when the deals will be put in concrete form.

Sightseeing

Toulouse town, the place of congress is such an historical site that you must not leave this beautiful *pink city* without visiting it.

Other places distant of Toulouse from a hundredth of miles are very exciting, for example the fortress of Carcassonne and also a lot of landscapes or villages in the Pyrénées mountains. Have a good trip in the Midi-Pyrénées and Languedoc-Roussillon districts.

Announcement: SGML & T_EX Conference, Groningen, 31 Aug. 1990

Organizing Committee:

NTG: C.G. van der Laan

T.A. Jurriens

SGML: J. Maasdam

J. Bleeker

This conference, devoted to SGML, T_EX, and their symbiosis, is organized by the NTG (Nederlandstalige T_EX Gebruikersgroep, i.e., Dutch T_EX Users Group) and the SGML-Holland Users Group. The conference will be held at the University of Groningen.

After some coffee, welcome etc., the day will start with a survey (invited speaker) about the relationship between SGML and T_EX, now and in the future. During the day we have two parallel streams of presentations (1/2 and 3/4 hour talks) and for those wandering around there will be a vendor booth, a book stand, a selling point for T_EX ‘gadgets’, and copying facilities for PD (T_EX) programs, and of course an information booth. The closing session (invited speaker) will be about Electronic Publishing in the future with a wink to SGML and T_EX.

To celebrate the joint happening we will end with a nice cocktail party offered by Elsevier and Samson Publishers.

The conference language is Dutch (this does not hold of course for non-native speakers) but speakers will be urged to have their transparencies in English. T_EX, L^AT_EX, Metafont (TUG-like) courses as well as SGML courses are held before the conference and in the week after.

Members of any SGML or T_EX Users Group are charged for f75-, students for f25-, and non-members f150,-.

Further information: C.G. van der Laan (CGL@RC.RUG.NL)
RekenCentrum RijksUniversiteit Groningen
Landleven 1, 9700 AV
Groningen, The Netherlands

NTG-SGML

Conference

Conference Booking Form

SGML & T_EX conference

Groningen, 31 Aug. 1990

For members of NTG, SGML Holland, DANTE, GUTenberg, NordicTuG, TUG, UKTuG, there will be a charge for the meeting of *f* 75,-, to cover refreshments, lunch etc. For non-members the fee is *f* 150,-, except for students who have to pay *f* 25,-. (Student card must be shown at registration on 31st August.)

I wish to attend the one-day SGML-T_EX meeting at the University of Groningen on 31st August 1990:

Name: _____

Organization: _____

Address: _____

Telephone: _____ Email: _____

Payment: *f* 25,- Student *f* 75,- NTG/SGML/member any SGML or T_EX Users Group *f* 150,- Non-memberCheck: enclosed

Signature: _____

 send separately

Date: _____

Payment and Registration Address Please send this completed form, together with your payment, to:

Koen Mulder: Penningmeester SGML Holland Users Group

Postbus 1148, 6801 MK, Arnhem, The Netherlands

Giro: 2905269

Note: How to come there will be send in July.

Adressen

DANTE

Deutschsprachige Anwendervereinigung T_EX e.V.

Rechenzentrum der Universität Heidelberg

Im Neuenheimer Feld 293

D-6900 Heidelberg 1

e-mail: dante@dhdurz1

Vorstand:

Joachim Lammarsch 1. Vorsitzender

Gabriele Kruljac 2. Vorsitzende

Friedhelm Sowa Kassenswart

Luzia Dietsche Schriftführerin

T_EX Users Group

P.O. Box 9506

Providence, RI 02940

U.S.A.

e-mail: tug@math.ams.com

Autoren / Organisatoren
Thomas Birnthaler

Gerngroßstr. 13

D-8500 Nürnberg 10

Barbara Burr

Rechenzentrum der

Universität Stuttgart

Allmandring 30

D-7000 Stuttgart 80

zrfn0370@ds0rus52

Rainer Hülse,**Wolfgang Kaspar**

Rechenzentrum der

Westfälischen Wilhelms-Universität

Einsteinstr. 60

D-4400 Münster

urz76@dmswwu1a bzw.

urz86@dmswwu1a

Luzia Dietsche

Rechenzentrum der Universität

Im Neuenheimer Feld 293

D-6900 Heidelberg 1

rz68@dhdurz1

Gabriele Kruljac-Dronskowski

Max-Planck-Institute Stuttgart

Gemeinsame Datenverarbeitung

Heisenbergstr. 1

D-7000 Stuttgart 80

kruljac@ds0mpi11

Joachim Lammarsch

Rechenzentrum der Universität
Im Neuenheimer Feld 293
D-6900 Heidelberg 1
rz92@dhdurz1

MID

Logistics Information Group GmbH
Ringstr. 19
D-6900 Heidelberg 1
m_krueger@eurokom.ie

Pierre Legrand

C.I.C.T.
Univ. Paul Sabatier
118, Route des Narbonne
F31062 Toulouse Cedex
legrand@frcict81

Konrad Neuwirth

Fernkorngasse 44/2/4
A-1100 Wien, Österreich
a4422dae@awiuni11.bitnet

Hubert Partl

EDV Zentrum der
Technischen Universität Wien
Wiedner Hauptstr. 8-10
A-1040 Wien
z3000pa@awituw01

Peter Scherber

Gesellschaft für
wiss. Datenverarbeitung mbH
Am Faßberg
D-3400 Göttingen-Nikolausberg
zscherb@dgogwdg1

Christina Thiele

JPC DT1711
Carleton University
Ottawa, Canada K1S 5B6
FAX: 613-788-3544
wsscat@carleton.ca

Drs. C.G. van der Laan

Computing Centre
P.O. Box 800
NL-9700 AV Groningen
clg@hgrrug5

Site-Koordinatoren

- AIX Uwe Untermarzoner
Kirchentellinsfurterstr. 3
7400 Tübingen 9
Tel: 0711/7207-4099
e-mail: untermar@dhdibm1
- Amiga Konrad Neuwirth
Fernkorngasse 44-2-4
A-1100 Wien
Tel: +43/222/6041530
e-mail: a4422dae@awiuni11
- Atari Stefan Lindner
Iltisstr. 3
D-8510 Fürth
Tel: 0911/7591886
oder
Lutz Birkhahn
Fürtherstr. 6
D-8501 Cadolzburg 2
Tel: 09103/2886
- BS2000 Friedhelm Sowa
& PC Heinr.-Heine Universität
Rechenzentrum
Universitätsstr. 1
D-4000 Düsseldorf
Tel: 0211/3113913
e-mail: tex@dd0rud81
- Mac Lothar Meyer-Lerbs
Am Rüten 100
D-2800 Bremen 33
Tel: 0421/252624
e-mail: g07m@dhbrrrz41
- MVS Ferdinand Hommes
GMD Z1.BN
Riemenschneiderstraße 11
5300 Bonn 2
Tel: 0228/8199621
e-mail: grztex@dbngmd21
- NOS/VE Norbert Schwarz
METAFONT Ruhr Universität
Rechenzentrum
Universitätsstr. 150
D-4630 Bochum
Tel: 0234/700-3940
e-mail: p020021@dborub01
- UNIX Dipl.-Inf. Karl Kleine
FZI Karlsruhe
Haid-und-Neustr. 10-14
D-7500 Karlsruhe 1
Tel: 0721/69060
e-mail: kleine@ira.uka.de
- VM/CMS Dr. Georg Bayer
TU Braunschweig,
Rechenzentrum
Postfach 3329
D-3300 Braunschweig
e-mail: c0030001@dbstu1
- VMS ??? gesucht !!!
Kandidaten bitte bei Herrn
Lammarsch melden!
- Treiber Joachim Schrod
L^AT_EX Kranichweg 1
D-6074
Rödermark-Urberach
Tel: 06074/1617
e-mail: xitijsch@ddathd21
- L^AT_EX- Hubert Partl
Styles EDV Zentrum der
Technischen Universität Wi-
en
Wiedner Hauptstr. 8-10
A-1040 Wien
e-mail: z3000pa@awituw01

Inhalt Heft 1/90

Impressum	2
Editorial	3
Hinter der Bühne	4
Protokoll der 2. Mitgliederversammlung von DANTE e.V.	4
Von fremden Bühnen	12
Deutsche Kurzfassungen: TUGboat-Artikel Vol. 11, No. 1	12
Die Bretter, die die Welt bedeuten	16
The New Versions of T _E X and METAFONT	16
BIB _T E _X und MAKEINDEX	23
Setzen von Regeln mit <code>regel.sty</code>	24
Struktogramme setzen mit <code>strukto.sty</code>	27
Bearbeitung mehrteiliger L ^A T _E X-Dokumente mit <code>TEXPART.BAT</code>	32
EXPDLIST – eine Erweiterung der <code>description</code> -Umgebung	34
Rezensionen	37
x _T E _X , y _T E _X , z _T E _X , Wieviele weitere Versionen wird es noch geben?	37
T_EX-Beiprogramm	40
Publisher	40
Magazin	42
Großer Ideen-Wettbewerb	42
Bücherecke	42
T _E X-Poesie	44
Der Fortsetzungsroman	46
Spielplan	47
Tagungstermine	47
Gutenberg'90	48
Announcement: SGML & T _E X Conference, Groningen, 31 Aug. 1990	51
Adressen	53
Autoren / Organisatoren	53
Site-Koordinatoren	55