

Die T_EXnische Komödie

dante

Deutschsprachige
Anwendervereinigung T_EX e.V.

17. Jahrgang Heft 3/2005 August 2005

3/2005

Impressum

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

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

Diese Ausgabe wurde mit Hilfe folgender Programme erstellt: **pdf_eTeXk**, **Version 3.141592-1.21a-2.2 (Web2C 7.5.3)**. Als Standard-Schriften kamen die Type-1-Fonts Latin-Modern und LuxiMono zum Einsatz.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 2700

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

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

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

Redaktion: Herbert Voß (verantwortlicher Redakteur)

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

Redaktionsschluss für Heft 4/2005: 15. Oktober 2005

ISSN 1434-5897

Die T_EXnische Komödie 3/2005

Editorial

Liebe Leserinnen und Leser,

etwas zeitverzögert, aber dafür gleich im Doppelpack erscheinen Berichte zu den T_EX-Theatertagen, der EuroT_EX 2005. Diese zeigen Ihnen, welche »Vorstellungen« es auf der »großen europäischen T_EX-Bühne« gab. Die Veranstaltungen des lokalen »Theaters« sind natürlich genauso wichtig, weshalb Sie in diesem Heft auch die Einladung zur Herbsttagung von DANTE e.V. im hohen Norden finden und sich diesen Termin schon einmal vormerken sollten.

Ein lange geplanter Artikel zum Thema Schach ist nun endlich fertig geworden und zeigt, dass man mit T_EX auch dieses Feld erfolgreich bespielen kann. »Live CDs« haben nicht nur auf T_EX-Bühnen eine gewisse Aufmerksamkeit erlangt. Der Artikel beschreibt, was alles bei dem Einsatz derartiger Versionen zu beachten ist. Zu beachten gibt es auch vieles, wenn es an das Konvertieren von Texten von und nach L^AT_EX geht. Die dazu gesammelten Erfahrungen mit `writer2latex` können Sie in einem Beitrag nachlesen.

Zwei weitere Artikel beschäftigen sich zum einen mit $\epsilon\lambda$ T_EX und zum anderen mit dem `xkeyval`-Paket. Ersterer ist ein Zwischenbericht eines langen Weges, der hoffentlich nicht das Ziel bleibt, sondern irgendwann zum selbigen führt. Viele Pakete verwenden mittlerweile die `keyval`-Notation zur Übergabe von optionalen Parametern an Makros. Mit dem `xkeyval`-Paket lässt sich die Anwendung noch einmal erweitern und insbesondere für größere Projekte viel besser handhaben. Doch lesen Sie lieber selbst ...

Mit T_EXnischen Grüßen,

Ihr Herbert Voß

Hinter der Bühne

Vereinsinternes

Grußwort

Liebe Mitglieder,

seit dem letzten Grußwort fanden zwei wichtige Ereignisse statt.

Beide Unterzeichner nahmen an der Practical $\text{\TeX}$ vom 14. bis 17. Juni in Chapel Hill (North Carolina, USA) teil und waren dort auch mit Vorträgen vertreten. Dies war die zweite Konferenz dieses Namens nach der Practical $\text{\TeX}$ 2004 in San Francisco. Dem Titel entsprechend widmete sich die Practical $\text{\TeX}$ der Nutzung von $\text{\TeX}$, $\text{\LaTeX}$, $\text{\ConTeXt}$ und Co. in der Praxis und stellte somit ein ergänzendes Angebot der TUG für $\text{\TeX}$ -Nutzer im amerikanischen Raum dar, da die diesjährige jährliche TUG-Konferenz wiederum außerhalb Amerikas stattfindet.

50 Teilnehmer erlebten zunächst eintägige Tutorien – wahlweise $\text{\TeX}$ für das WWW (Peter Flynn), Einführung in $\text{\ConTeXt}$ (Steve Peter) oder $\text{\LaTeX}$ für Anfänger (Cheryl Ponchin). An den restlichen drei Tagen gab es Vorträge zu einem breiten Themenbereich, nach einer Retrospektive von Nelson Beebe über die Entwicklung von $\text{\TeX}$ und $\text{\METAFONT}$ zunächst einen rege diskutierten Vortrag über $\text{\LaTeX}$ aus dem Blickwinkel eines Anfängers (Peter Flom), dann mehrere Vorträge über verschiedene Wege zum Erstellen von Beamer-Präsentationen mit $\text{\LaTeX}$: beamer (Andrew Mertz und William Slough), $\text{\TeX}$ power und PSTricks (David Allen) und HA-Prosper (Tristan Miller). Weitere Themen waren MathML und $\text{\TeX}$ 4ht (Eitan Gurari), das Einbinden von Grafiken (Klaus Höppner), $\text{\LaTeX}$ und PitStop (Mirko Janc) und $\text{\TeX}$ und XML (Hans Hagen).

Auch für Berichte vom Einsatz von $\text{\TeX}$ in der Praxis war genügend Raum, seien es das Erstellen kritischer Editionen (John Burt), das Zusammenspiel von makeindex mit der Analysesoftware SAS (Ronald Fehd), die Migration von Word zu $\text{\TeX}$ bei der Internationalen Atomenergiebehörde in Wien (David Igant), $\text{\TeX}$ im Einsatz beim Erstellen von Konferenzbänden aus XML-

Daten (Volker RW Schaa) bis hin zum Erstellen einer Broschüre über die Pflanzen im Zoo von Los Angeles (Joseph Hogg).

Wie viele andere Konferenzen diente die Practical T_EX auch dem Gedankenaustausch der Teilnehmer neben den Vorträgen, insbesondere in den Kneipen und Restaurants von Chapel Hill, einer netten kleinen Universitätsstadt, die wenig in die Vorstellungen von den typisch amerikanischen Südstaaten passte. Beide Unterzeichner haben die Treffen mit vielen amerikanischen Freunden sehr genossen. Einziger Wermutstropfen war der Kampf mit den amerikanischen Klimaanlagen. Während außen Temperaturen jenseits der 30 Grad herrschten, wünschte man sich im Innern eher Winterbekleidung herbei.

Im Laufe der Konferenz fand auch ein Treffen der anwesenden TUG-Mitglieder statt. Hier wurde das Ergebnis der Präsidentenwahl der TUG offiziell bekannt gegeben. Nach langer Zeit standen wieder zwei Kandidaten für das Amt des TUG-Präsidenten zur Wahl – der bisherige Amtsinhaber Karl Berry und Lance Carnes, vielen bekannt als Inhaber von Personal T_EX Inc., dem Hersteller des kommerziellen PCT_EX. Nach Auszählung der eingegangenen Briefwahlzettel ergab sich ein Vorsprung von 6 Stimmen für Karl Berry, der somit die Chance einer zweiten Amtszeit erhält und dem wir an dieser Stelle herzlich gratulieren. Seine Aufgabe wird nicht einfach sein, da die TUG seit Jahren unter einem stetigen Schwund von Mitgliedern leidet und dies natürlich zu einer angespannten Finanzsituation führt.

Das zweite große Ereignis war der Stand von DANTE e.V. beim Linuxtag in Karlsruhe (22.–25. Juni). Er konnte viele Besucher anziehen, die sich für T_EX interessierten. Mit Hilfe von Postern und mitgebrachten PCs wurden viele Praxisbeispiele für den Einsatz von T_EX vorgeführt, und die freiwilligen Helfer mussten viele Fragen beantworten (und brachten nebenbei 22 Schnuppermitgliedschaften an den Mann und die Frau). Umrandet war das Ganze von einem Vortragsprogramm, sei es zu L_AT_EX, Beamerpräsentationen oder Emacs mit AucT_EX. Wir möchten an dieser Stelle allen, die zum Gelingen des Stands beigetragen haben, unseren herzlichen Dank aussprechen. Karlheinz Geyer und andere haben netterweise für eine der zukünftigen Ausgaben von »Die T_EXnische Komödie« einen Artikel über die Erfahrungen und Erlebnisse angekündigt.

Mittlerweile haben sich für die Herbsttagung von DANTE e.V. Ort und Zeit gefunden. Sie wird am 29. und 30. September 2005 am Institut für Informatik und Praktische Mathematik der Universität Kiel stattfinden. In dieser Ausgabe von »Die T_EXnische Komödie« finden Sie den »Call for Papers« und

die Einladung zur Mitgliederversammlung. Auf der Mitgliederversammlung soll das Redaktionsstatut von »Die $\text{T}_{\text{E}}\text{X}$ nische Komödie« geändert werden, um zukünftig die Veröffentlichung zurückliegender Ausgaben im WWW zu vereinfachen.

Vermutlich zeitgleich zum Erscheinen dieser Ausgabe wird in China die TUG 2005 stattfinden. Als zukünftige Orte für TUG-Konferenzen sind Marokko (ca. November 2006) und San Diego (2007) geplant. Den Lesern, die sich eher für europäische Tagungen interessieren, möchten wir das diesjährige Treffen der italienischen Gruppe GuIT ans Herz legen, das am 22. Oktober 2005 in Pisa stattfindet.

Mit freundlichem Gruß,

Volker RW Schaa	Klaus Höppner
Vorsitzender	Stellvertretender Vorsitzender

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

Volker RW Schaa, Karsten Heymann

Liebe Mitglieder von DANTE e.V.,

die nächste $\text{T}_{\text{E}}\text{X}$ -Tagung von DANTE e.V. findet am 29. und 30. September 2005 in der Christian-Albrechts-Universität zu Kiel statt. Veranstalter ist DANTE e.V. mit Unterstützung des Instituts für Informatik und Praktische Mathematik der Christian-Albrechts-Universität Kiel.

Für Donnerstag sind Tutorien und Vorträge geplant, am Freitag wird die 33. Mitgliederversammlung von DANTE e.V. stattfinden. Die Tagesordnung der Mitgliederversammlung am 30. 9. 2005 um 9.00 Uhr in der

Christian-Albrechts-Universität Kiel
Audimax, Hörsaal C
Christian-Albrechts-Platz 2
24118 Kiel

lautet:

1. Begrüßung und Tagesordnung
2. Bericht des Vorstands
3. Änderung des Redaktionsstatuts von »Die T_EXnische Komödie«
4. Verschiedenes

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

Falls Sie ein Tutorium oder einen Vortrag anbieten wollen, werden Sie gebeten, dies mit dem Anmeldeformular unter <http://www.dante.de/dante/events/mv33/cfp/> oder per E-Mail an mv33@dante.de bei den Organisatoren möglichst bis 31. August 2005 anzumelden.

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

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

Die Homepage der Tagung findet sich unter <http://www.dante.de/dante/events/mv33/>.

Mit Fragen, Wünschen und Anregungen wenden Sie sich bitte an

DANTE e.V.

Stichwort: Mitgliederversammlung von DANTE e.V.

Postfach 10 18 40

69008 Heidelberg

E-Mail: mv33@dante.de

Mit freundlichen Grüßen,

Volker RW Schaa (DANTE e.V.)

Karsten Heymann (Kiel)

T_EX-Theatertage

2² – DANTE e.V. wird sechzehn

Erich Ruff¹

Dienstag Mittag. Vorbei war es mit der Tagungsruhe, dem entspannten Zuhören und sorgenfreien Dabeisein. Beim vorzüglichen Essen, von sehr aufmerksamen Bedienungen serviert, sprach mich Bernd Raichle darauf an, dass es jedes Jahr einen Tagungsbericht gegeben hätte. Die Bedeutung seines erwartungsvollen Blickes begann sich erst ganz langsam zu einer Ahnung und dann zur Gewissheit zu verdichten: er wollte mich mit dieser Aufgabe betrauen.

Sofort nach meinem »Auftrag« begann sich mein Blick auf die bereits vergangene Tagungszeit, auf die Anreise sowie auf die zukünftigen Veranstaltungen zu verändern und sich ein kritischer Filter einzuschalten. Was wohl von meinen persönlichen Eindrücken für die anderen Teilnehmer interessant sein könnte? Da es sich bei den meisten Teilnehmern um Naturwissenschaftler und Mathematiker handelte (der Unterschied sei für manche Mathematiker sehr wichtig, habe ich mir sagen lassen müssen), die mich immer wieder mit ihrem Fachwissen beeindruckt haben, kam ich mir als Psychologe schon oft als »irrationale Zahl« vor, die sich in den reellen Zahlenraum verirrt hat. Trotzdem wurde ich anfänglich irritiert, dann aber doch freundlich und interessiert aufgenommen.

Lange überlegte, verwarf, griff ich Ideen neu oder wieder auf und je näher der Redaktionsschluss rückte, desto größer wurde der Druck und desto ernsthafter befürchtete ich, dass ich meine Zusage würde zurückziehen müssen, bis mir beim Joggen die Idee kam, die Leser Gäste meiner Überlegungen und Gefühle während der Anreise und ansatzweise der Tagungszeit sein zu lassen.

¹Bedanken möchte ich mich bei Torsten Schütze. Er stand mir nicht nur mit seinem profunden L^AT_EX-Wissen zur Seite, sondern war auch ein gründlicher und kritischer Korrekturleser.

Wie immer vor einem mehrtägigen Schließen meiner psychotherapeutischen Praxis gab es mehr als die übliche Arbeit und damit Stress, dem ich durch eine zeitige Abreise ein Ende bereiten wollte, um die lange Fahrt als Erholung zu nutzen.

Ausgestattet mit Kaffee, Imbiss und einer Hörspiel-CD [9] freute ich mich – nach unerwartet dichtem Stadtverkehr – die Autobahn zu erreichen, um ohne Hektik im Verkehr mitschwimmend das Hörspiel zu genießen und mich meiner Vorfreude auf arbeitsfreie Tage, die Vorträge, die $\text{\TeX}$ ies und Frankreich zu überlassen. Das Hörspiel begann mich schnell zu fesseln und die Geschwindigkeitsbeschränkung ermöglichte es, meine Aufmerksamkeit auch anderen Dingen zuwenden zu können.

Ich hörte Selbs erste Worte:

Sie erinnerte mich an die Tochter, die ich mir manchmal gewünscht habe. Wache Augen, ein Mund, der gerne lacht, hohe Wangen und volle braune Locken bis auf die Schultern. [...] Seit Monaten sei die Familie ohne Nachrichten von Leonore. Ihr Vater hatte mich angerufen. [...] »Zuletzt haben wir noch gehofft, dass sie zu Semesterbeginn wieder auftaucht.« [...] »Ihre Tochter hat sich bei der Universität nicht wieder eingeschrieben?« Er antwortete gereizt: »Herr Selb, ich wende mich an einen privaten Ermittler, damit er ermittelt und nicht ich.« [9, S. 99]

Wie war das denn 1976? Es war die Zeit, als ich an der Bayerischen Akademie der Wissenschaften an der Akademieausgabe von J.G. Fichtes Werken [6] mitarbeitete. Der Verlag hatte immer größere Probleme eine Setzerei zu finden, die – wie unser Chef es sich wünschte – noch in Blei setzte. Wie wäre das gewesen, wenn wir 1984, als das $\text{\TeX}$ book [3] von Donald Knuth erschien, begonnen hätten mit $\text{\TeX}$ zu arbeiten? Der komplexe, wissenschaftlich editierte Satz benutzt verschiedene Schriftschnitte und Schriften – auch Fraktur – zur Kennzeichnung von unsicheren Lesarten, von Einfügungen der Herausgeber, von Unterschieden, die sich in verschiedenen früheren Werkausgaben finden usw. Die PostScript-Fonts hätte die Kommission erst ab 1984 kaufen können, aber wer hätte deren Anpassung übernommen, die Ligaturen nachgearbeitet, das Kerning – vor allem bei der Variante »kursiv-gesperrt« – optimiert? Einen Yannis Haralambous, der die Frakturschriften 1991 erstellte, gab es damals in der $\text{\TeX}$ -Welt ebenso wenig wie die vorgefertigten Fontmetriken von Walter Schmidt.

Der erste Stau bei Augsburg riss mich aus meinen Gedanken und Selbst sprach gerade:

Beim Baggersee vor Sankt Ilgen hielt ich an, stieg aus und trat ans Ufer. Ich versuchte, Kieselsteine übers Wasser hüpfen zu lassen. Es ist mir schon als Junge am Wannsee nicht geglückt. Ich werde es auch nicht mehr lernen. Darum lasse ich mir von einem jungen Bürschchen in weißem Kittel aber noch lange nichts vormachen. Wendts Geschichte war faul. Wo war die Polizei geblieben? Eine junge Frau ist seit drei Monaten im Psychiatrischen Landeskrankenhaus, stürzt aus dem schlecht gesicherten dritten Stockwerk, und niemand denkt an fahrlässige Tötung oder Schlimmeres und holt die Polizei?

[9, S. 38]

Ich hatte den Faden verloren. Mit dem Zurückspringen auf wieder bekannte Passagen gelang es mir in Erfahrung zu bringen, dass

Ich wußte, wo Leo gewohnt, studiert und verkehrt hatte [...] Seit Januar war sie verschwunden, [...] Im Juli letzten Jahres war sie von der Polizei festgenommen und erkennungsdienstlich behandelt worden [...] das Psychiatrische Landeskrankenhaus vor den Toren Heidelbergs. Ich hatte es nicht vergessen. Ich habe mich davor gedrückt. [...] Ja, Frau Salger war Patientin bei uns. [...] Ich wartete. »Herr Dr. Wendt?« [...] Er setzte ein paarmal an. »Ich kann Ihnen ... muß Ihnen ... Frau Salger ist tot. [...] Ein tragischer Unglücksfall.«

[9, S. 30 ff.]

Der flüssig werdende Verkehr sowie die reichlich vorhandene Zeit ließen schnell wieder ein entspanntes Urlaubsgefühl sich breit machen und Selbst fesselte meine Aufmerksamkeit, als er zu Dr. Wendt sprach:

»Und wenn ich jetzt nicht gekommen wäre, hätten Sie sie beerdigen lassen, ohne daß die Eltern auch nur ein Sterbenswörtchen davon erfahren? Was für eine verrückte Geschichte erzählen Sie mir denn da, Herr Dr. Wendt!«

[9, S. 36]

Wie wäre die Registerhaltigkeit zu erzielen gewesen, die im Bleisatz so gut wie möglich zu gewährleisten gesucht wurde? Die ersten Proben des damals versuchsweise eingesetzten Lichtsatzes hielten das Register auch nicht hundertprozentig. Diese Aufgabe mit T_EX zu bewältigen wäre nur mit großem Programmieraufwand möglich gewesen. Wieviel da zu leisten gewesen wäre, das kann Hans Hagen sicher treffend beantworten. Aber bis Hans Hagens

ConT_EXt diese Möglichkeit zur Verfügung stellen würde, hätten noch viele Jahre vergehen müssen.

Der Satzspiegel und die Titelei wären machbar gewesen, aber welche Rechner hätten wir damals überhaupt benutzen können?

Die erste L^AT_EX-Distribution, an die ich mich erinnern kann, bestand aus über 10 Floppies (5¹/₂ Zoll), die ich 1986 auf meinem gebrauchten, für teures Geld erstandenen (100% IBM-kompatiblen) XT nicht zum Laufen bringen konnte. Unsere »Rechner«, auf denen wir die im 18. Jahrhundert mit Tinte und Gänsekiel geschriebenen Texte Fichtes für alle leserlich herstellten, waren mechanische Schreibmaschinen. Mein Chef »flog« auf einer alten Adler, meine Kollegen behackten jüngere Modelle und meine IBM-Kugelkopf stellte die Avantgarde des technisch eingesetzten Materials dar. Eine T_EX-Installation wäre wohl nur im Leibniz-Rechenzentrum verfügbar oder machbar gewesen. Aber sicher bin ich mir nicht. 1975, während meines Soziologiestudiums, habe ich noch Hollerithkarten gelocht, um die Daten unserer Umfrage statistisch auszuwerten.

Jedenfalls, so gemütlich wie ich damals zu Hause – immer eine Tasse Tee griffbereit – am Fichte arbeitete, meine Dissertation weiter entwickelte und mit meinem Freund und Arbeitskollegen unser erstes Buch »Der begriffene Wahnsinn« [10] schrieb, wäre es im LRZ sicher nicht gewesen. Sebastian Rahtz und die anderen »guten Geister«, die die T_EX Live als ein »überalldabei«-System entwickelten, gaben dem Schreiben von Text mit einer Reiseschreibmaschine jedenfalls eine ganz andere Bedeutung, als die, die sie für meine Kollegen und mich damals hatte.

Stuttgart kündigten die Hinweisschilder an, und ich freute mich auf das Ende der Geschwindigkeitsbeschränkung, obwohl der immer noch dichte Verkehr Schnellfahren nicht zulassen würde.

Selb hatte inzwischen einen »Wärter« in der Psychiatrie nach dem Unfall befragt. Dieser sah ihn verständnislos an und fragte Selb: »Wer hat ihnen denn den Bären aufgebunden?« [9, S. 44] Er brachte in Erfahrung, dass auch die Polizei nichts von dem »Unfall« wusste. Er durchsuchte die Wohnung von Dr. Wendt, fand außer drei Telefonnummern nichts Erfolg versprechendes oder Bedeutungsvolles. An einer der Nummern meldete sich eine Frau mit »dickem amerikanischem Akzent«, die aber deutsch zählte. Dann informierte er sich schließlich in der Stadtbücherei über Psychiatrie:

Ich fand geistige Gesundheit als die Fähigkeit definiert, das soziale Spiel gut zu spielen. Geistig krank ist, wen wir nicht mehr ernst nehmen, weil er nicht oder schlecht mitspielt – mir kroch es kalt den Rücken hoch. [9, S. 68]

Mir auch. Unsere Überlegungen zu der Frage, was geschieht eigentlich mit einem Menschen, der verrückt wird, und was haben die Bewusstseinstheoretiker Fichte, Kant und Hegel dazu zu sagen, fiel mir wieder ein. Als wir nichts Aussagekräftiges fanden, begann unsere theoretische Beschäftigung mit dem Wahnsinn, und bald darauf machten wir unsere ersten praktischen Erfahrungen mit Patienten und der Einrichtung Psychiatrie. Unser erstes Buch »Der begriffene Wahnsinn« [10] entstand aus dieser Beschäftigung. Wir ließen es setzen. Mit Wordstar wurde es 1984 von unserem Schreibmaschinenmanuskript abgeschrieben. Deshalb gab es keine Anmerkungen und Fußnoten, obwohl wir ursprünglich genug davon im Manuskript gehabt hatten. Es ginge einfach nicht, sagte die Dame vom Schreibbüro. Auch die Seitenzahlen konnte sie nicht mittig setzen. Wir fügten uns und hatten ein wissenschaftliches Buch, das den Lesern in Romanform gegenübertrat. Für ein Buch über den Wahnsinn gar nicht schlecht, zwingt es doch den Leser in einen unauflöselichen Widerspruch zwischen Form und Inhalt.

Beim Fichte wäre diese Vereinfachung nicht möglich gewesen. Was man heute an komplexen Texten und Katalogen sogar automatisch setzen kann, davon würde Stephan Lehmke auf der Tagung berichten. Wie aber hätten wir den Fichte in $\text{\TeX}$ setzen können, mit zum Teil drei Anmerkungsapparaten, dessen einer in alphabetisch geordneten Kleinbuchstaben Lese- bzw. Druckvarianten verzeichnete, während der zweite numerisch fortlaufend die Literaturverweise auflistete. Im dritten folgen – mit Sternchen nummeriert und in normaler Schrift – Fichtes Notizen, die sich neben, über oder unter dem Haupttext auf dem Blatt befanden. Die dazu notwendigen Informationen bereichern ebenfalls die beiden erstgenannten Anmerkungsapparate.

Die Aufgabe, einen solchen Anmerkungsapparat automatisch zu verwalten, wäre nur unter größten Schwierigkeiten zu lösen gewesen. In welchen komplexen Problemen wir gekommen wären, zeigte David Kastrup schon in Hannover. Donald Knuth hat $\text{\TeX}$ wohl wirklich »nur« für Mathematiker und Naturwissenschaftler (sic!) entwickelt, die Geisteswissenschaftler blieben erst einmal außen vor.

Selb erfuhr inzwischen vom Direktor der psychiatrischen Klinik, dass

es tönicht von Wendt [war], einen tödlichen Unfall zu erfinden. [...] Wendt hat einfach die nächste dumme Lüge genommen, um sie abzuwimmeln und seine Klientin abzuschirmen. »Und wo ist sie?« »Herr Selb, ich weiß es nicht.« [9, S. 72]

Inzwischen hatte ich wieder einmal den Faden verloren, musste so lange zurückspringen auf der CD, bis ich wieder an der Stelle war, wo ihm auffiel, dass sein Freund Giovanni italienisch zählte. Danach suchte er die Adresse auf, die zu der Telefonnummer gehörte, an der sich die junge Frau mit dem »dicken amerikanischen Akzent« gemeldet hatte.

[dann] hatte ich sie voll im Blick, und es blieb kein Zweifel. Die junge Frau war Leo. Leo mit rosaroter Sonnenbrille, wasserstoffsuperoxyd-blondem Lockenkopf und kariertem Männerhemd über den Jeans. [9, S. 75]

Nun ermittelte Selb, dass Leos Vater in heutigen Regierungskreisen nicht als Ministerialdirigent bekannt war und eigentlich Lehmann hieß. Er machte die alkoholranke Mutter von Leo ausfindig, deren Mann schon lange gestorben war. Er war in eine Unterschlagungsgeschichte als Ministerialdirigent verwickelt, in Haft genommen, frei gelassen worden und schon lange an Krebs gestorben.

Wo ist denn jetzt das Ende der Geschwindigkeitsbeschränkung? Am Dreieck Leonberg war ich gerade vorbei, aber 120 mahnten die Verkehrsschilder immer noch. Das wird doch nicht bis hinunter nach Karlsruhe so gehen, dachte ich noch, als mich Selb wieder fesselte. Der »tote« Salger rief ihn an. Nach langen Verhandlungen war er

den Fall Salger leid. [...] Ich kann meinen Bericht auch [...] c/o Helmut Lehmann [schicken]. Einen Moment war es still in der Leitung. Dann knallte Salger den Hörer auf die Gabel. [9, S. 100]

Mein immer noch gut bestücktes Zeitkonto mir vergegenwärtigend, reihte ich mich entspannt auf der rechten Spur ein.

Seit 2²² Jahren gibt es DANTE e.V. nun schon, schweiften meine Gedanken wieder ab von Straße und Hörbuch. Heute kann ich die T_EX Live offline oder CTAN online nach passenden Styles, Vorarbeiten oder Dokumentationen durchsuchen. Was hätte ich 1984 gemacht? Der Kopka [4] erschien 1988, der »Companion« [1] 1993 und der »Lamport« 1994 [5]. Den Fichte mit T_EX zu setzen, das wäre eine Aufgabe für ein ganzes Entwicklerteam gewesen.

Meine philosophischen Arbeitskollegen, fiel mir ein, scheuten damals vor der Anschaffung elektrischer Schreibmaschinen zurück. Wie hoch wäre wohl die Schwelle gewesen, die sie zu überwinden gehabt hätten, um sich $\text{\TeX}$ anzueignen?

Also Fichte und $\text{\LaTeX}$, das wurde mir immer klarer, das wäre damals nicht möglich gewesen und würde selbst heute noch Programmieranforderungen stellen, die nicht trivial sein würden.

Nun endlich, in der schnellen Fahrt bis Straßburg – während der sich Selb ausruhen konnte – lösten drängelnde Dieselfahrer, die vorbei wollten, aber diese Absicht mangels Leistung nicht umsetzen konnten, schlafende Sonntagsfahrer ab, die durch blinde Spurwechsel energisches Bremsen nötig machten. Das muss jetzt schon Frankreich sein, wo ist denn die Grenze gewesen? Und schon umgab mich wieder dichter Verkehr und die Suche nach der richtigen Ausfahrt nahm meine ganze Aufmerksamkeit in Anspruch.

Auf der »route nationale« empfing mich nach kurzer Zeit französische Atmosphäre. Häuser und kleine Ortschaften bezauberten mit ihrem Charme, den sie nicht aus deutscher Ordentlichkeit, sondern aus französischer Unbekümmertheit und Nachlässigkeit bezogen. Sie verlockten zum Spazierengehen und Photographieren, aber wollte ich zum Abendessen ankommen, musste ich zügig weiterfahren und auch einmal tanken. Das Benzin in Frankreich ist ja viel billiger – so hatte mich zu Hause ein kurzer Blick ins Internet belehrt – und der Plan, hier mit leerem Tank anzukommen, hatte sich gut umsetzen lassen.

Ich startete mein Hörbuch erneut, ließ Selb weiter ermitteln und hielt nach einer Tankstelle Ausschau. Die erste erschreckte mich mit ihrem überhöhten Preis, die nächste war noch teurer. Irritiert entschloss ich mich, bis zu einer billigen weiterzufahren. Aber weit und breit war auf einmal keine Ortschaft mehr, und bald machte mich mein Auto mit dezentem Piepsen darauf aufmerksam, dass der Tankinhalt noch für etwa 50 Kilometer reichen würde. Meine Überlegungen begannen sich von Konkordanzproblemen auf die Frage zu verlagern, was ich mit leerem Tank am späten Sonntagnachmittag in Frankreichs »Niemandland« machen würde. Meine besorgten Gedanken verfliegen im Nu und wichen einer tiefen Erleichterung. Die rettende Tankstelle war da, aber sie war vollautomatisch und nahm nur französische Kreditkarten. Soll ich hier irgendwo übernachten und morgen weiterfahren? Aber dadurch wird die Entfernung zur nächsten Zapfsäule auch nicht geringer. Also begann ich an Gefällestrcken auszukuppeln, um meinen hungrigen Motor auf Diät zu setzen. Schon 40 Kilometer auf Reserve ... viel geht jetzt nicht

mehr. Was hilft jetzt High Tech, Handy, Computer und der schönste Blocksatz. Begann meine sich anschleichende Verzweiflung die Errungenschaften der modernen Technik zu entwerten? Die blödsinnige Idee, die deutsche Leidenschaft, ein paar Euro am Sprit zu sparen, wurde mir in ihrer ganzen Absurdität bewusst, und dass Reisen bildet, bekam plötzlich einen Sinn, der sich nicht mehr nur auf Kulturgüter, sondern auf idiotisches Alltagsverhalten bezog.

Ist das wirklich eine Tankstelle? Sie war geschlossen, aber an der Ortsausfahrt gibt es noch eine, wies mir ein Schild. Ob das noch gut gehen wird? Zwei gemächlich schaltende Ampeln taten das ihrige, meine Spannung noch zu steigern, und dann stand ich endlich vor der Zapfsäule und konnte es erst nicht glauben. Der Preis war mir gleichgültig und auf der Rechnung zeigte sich, dass der Tank wohl bis auf einen knappen Liter leer gefahren war.

Erleichtert und entspannt genoss ich nun die gut ausgebaute Straße, kehrte zurück zu einem flotteren Fahrstil und musste mein Hörbuch weit zurücksputzen.

Selb ging in sein neues Stammcafe und ein anderer Gast setzte sich ihm gegenüber.

Ich erkannte ihn an der Stimme, noch ehe er sich mit »Salger« vorstellte. [...] Herr Selb, ich bin Ihnen eine Erklärung schuldig. [...] Wissen Sie, ich kenne Leonore Salger seit vielen, vielen Jahren. [...] Aber großer Bruder und kleine Schwester, altersmäßig weit auseinander, innerlich eng verbunden – Sie wissen, welche Konstellation ich meine? [...] Wo ist Leo? [...] Ich stand auf und ging. [9, S. 103 ff.]

Wie sollte der Krimi jetzt weitergehen, fragte ich mich. Das Buch hat doch über 200 Seiten. Muss ganz schön kompliziert sein, eine solche Handlung zu entwerfen, ging mir durch den Kopf. Aber Kompliziertheit entsteht schnell, will man einer Sache gerecht werden. Die Anmerkungen bei der Fichte-Ausgabe sind ein schönes Beispiel dafür. Der mit Kleinbuchstaben laufende Anmerkungsapparat wechselte, wenn er beim *z* angekommen war, zum *a*, das ein Strichlein bekam, wie alle folgenden Buchstaben. Nach dem *z'* kam dann wieder ein ungeschmücktes Alphabet. Mussten weitere Anmerkungen nachträglich eingefügt werden, gab es oft mehrere Strichlein. Wurde deren Zahl zu groß, so wurden umfangreiche Änderungen am Umbruch notwendig. Je mehr Seiten der Setzer aufbinden musste, desto mehr Korrekturlesegänge wurden erneut nötig. Dieses Problem – den bereits gesetzten Text zu erhal-

ten – ist mit T_EX aus der Welt geschafft. Aber wie wäre die alphabetische Neusortierung in T_EX zu programmieren gewesen? Der korrekte Umbruch hätte über mehrere Seiten iteriert werden müssen, damit zusammenbleibt oder -kommt, was zusammengehört. Ob das ϵ X_TE_X-Team dieses schwierige Problem angeht oder gar löst? Bin gespannt, was Gerd Neugebauer und Michael Niedermair dazu sagen werden.

Das Stichwortregister fiel mir als nächste Baustelle ein. Fichte schrieb seine Texte mit der Unbekümmertheit, die in dieser Zeit üblich war. Es gab keine Rechtschreibregeln. Jeder schrieb so, wie er meinte. Wichtige Begriffe finden sich oft groß geschrieben, das Wort Bewusstsein konnte einmal ein scharfes »s« haben, das andere Mal ein einfaches. Das »y« alternierte zufällig mit dem »i«, das Urtheil schrieb sich mal mit, mal ohne »h«. Das Register aber fasste alle Schreibweisen zu einer einzigen zusammen. Auch das wäre ein nichttriviales Problem gewesen, hätte es in T_EX gelöst werden sollen. Dabei fiel mir ein, dass ich auf den DANTE-Tagungen schon lange keinen Vortrag über Programmieren mit T_EX oder L^AT_EX mehr gehört habe. Heiko Oberdiek [7], erinnerte ich mich, sprach in Erlangen 2002 darüber, der ersten Tagung, an der ich teilnahm. Auch David Kastrup [2] trug über Programmierung vor, verirrte sich aber selbst in den Untiefen rekursiver Programmierung. Vielleicht könnte einer der vielen der Programmierung von L^AT_EX mächtigen DANTE-Mitglieder einen oder mehrere Vorträge darüber halten.

Was war in meinem Hörbuch in der Zwischenzeit geschehen?

Selbs Freund bei der Polizei, Nägelsbach, erzählte ihm, dass Dr. Wendt als Student zum Sozialistischen Patientenkollektiv gehörte. Als dieses sich auflöste, gingen einige Mitglieder zur RAF. Am Abend wurde Selb in seiner Wohnung von Salger erwartet, der ihn im Gesicht verletzte. Sein schönes Ledersofa musste zwei Einschüsse der mit einem Schalldämpfer bestückten Pistole hinnehmen, aber er leugnete standhaft, den Aufenthaltsort von Leo zu kennen.

Die Herren vom Bundeskriminalamt kamen am Morgen kurz nach acht. [...] Wir hören, daß Sie bei der Arbeit an einem Fall über Frau Leonore Salger gestolpert sind. Wir suchen Frau Salger. Wenn Sie ... »Warum suchen Sie Frau Salger?« »Das ist sozusagen eine heikle Frage.« [9, S. 123]

Plötzlich fand ich mich vor der Autobahnauffahrt nach Nancy und Metz wieder. Der lebhafte, von französischer Spontaneität geprägte Verkehr mit virtuosen Überholmanövern erforderte nun meine ganze Aufmerksamkeit.

Ich musste ausschalten, da ich nicht mehr länger zuhören konnte. Die Ausfahrt Pont-à-Mousson kam. Als ich vor der imposanten Abbaye parkte, dachte ich mir, dass dies ein würdiger Rahmen für den Auftritt von Donald Knuth und Hermann Zapf sein würde und dass sich hier 2²² Jahre DANTE e.V. wohl gut feiern lassen würden. Nach der langen Anreise und der Zeitreise durch 25 Jahre stieg ich erwartungsvoll und auch beruhigt aus, hatte ich doch bei der Fahrt zur Abbaye eine Tankstelle gesehen.

Die sehr freundlichen Empfangsdamen übergaben mir zu meiner großen Erleichterung den Schlüssel zu meinem Einzelzimmer, das ich schon im Herbst vorigen Jahres gebucht hatte. Die zweifache Aufforderung per E-Mail, mir einen Zimmergenossen auszusuchen, um eine »Zwangszusammenlegung« bzw. »-paarung« zu vermeiden, hing für mich immer noch in der Luft, zumal mir auf meine Beschwerden nur freundlich mitgeteilt worden war, dass »niemand übergangen werden würde«.

Von dem modernen Empfangsbereich trat ich durch eine Glastüre in die Abbay und fand mich in einem Gebäudekomplex, der gut als Kulisse für Harry Potters Schule Hogwarts gedient haben könnte. Leider war dem nicht so, denn als ich mit meinem Gepäck im zweiten Stock das Ende der engen, schmalen und schlecht beleuchteten Wendeltreppe erreichte, hing dort kein Bild wie bei Harry Potter.

At the very end of the corridor hung a portrait of a very fat woman in a pink silk dress.

“Password” she said.

“*Caput Draconis*” said Percy, and the portrait swung forward to reveal a round hole in the wall. [8, S. 142]

Es gab kein Bild, das ich durch ein Passwort, z. B. »*Caput Leonis*«, dazu bewegen konnte, die richtige Türe für mich zu öffnen. Also zurück zum Empfang und, nun genau informiert, kam ich über eine andere breite, wohl beleuchtete Treppe zu dem Gebäudeteil, wo tatsächlich mein Zimmer war.

Nach kurzem Auspacken wieder zurück beim Empfang, der nun mit anderen, lachend zusammenstehenden und redenden T_EXies besetzt war, traf ich

meinen Freund, der mir erzählte, dass auch er auf dem letzten Tropfen Diesel hier angekommen sei. Mit meinem schlechten Schulfranzösisch wollte ich ihn begleiten, damit ich den ADAC bzw. sein französisches Pendant rufen könnte, wenn wir unterwegs stehen bleiben würden. Das Lachen über unser Missgeschick verging uns immer wieder, weil sich die bange Frage bei jeder Ampel, bei jeder Abzweigung in den Vordergrund drängte, ob wir es noch bis zur Tankstelle schaffen würden. Als wir diese, kaum beleuchtet, fanden, mussten wir sie erst einmal – mit immer angespannteren Nerven – umkreisen, um die versteckte, dunkle Zufahrt zu finden. Dann standen wir vor der Zapfsäule und erst eine radebrechende Befragung der dort Tankenden erschloss uns die Logik dieses Systems, das Vorauszahlung verlangte. Wir gingen zum Zahlautomaten und bis wir alle unsere Kreditkarten versucht hatten, hatten wir hinter der Dieselpapfsäule einen ansehnlichen Stau produziert. Aber keine Karte war für den Automaten so vertrauenswürdig, dass er sie annehmen wollte. Als wir dann noch zwei junge Französinnen fragten, ob sie uns von ihrer Karte gegen Barerstattung abbuchen lassen würden, fanden wir uns beinahe wie Sittlichkeitsverbrecher behandelt und waren froh – ohne Polizei und ohne von den vielen wartenden Franzosen, die sich langsam um die Tankblockade geschart hatten, grob vertrieben zu werden – mit immer noch leerem Tank flüchten zu können.

Vor dem Hotel suchten wir einen abschleppgeeigneten Parkplatz und wurden im zu Fuß aufgesuchten Lokal mit einem mehrgängigen, sehr guten Diner entschädigt. Erst als es im Restaurant eigenartig ruhig wurde, stellten wir fest, dass die TeXies wieder einmal die letzten Gäste waren und machten uns in fröhlicher Stimmung auf den Heimweg.

Zum Begrüßungsvortrag erschien mein Freund mit erleichtertem Lächeln. Der Tank war voll und bis zur Pause hatten andere das Wort. Angenehm überrascht vom bereitgestellten Imbiss, tranken wir vom reichlich vorhandenen Kaffee, erfreuten uns am Kuchen und gegen Ende konnten wir an einem der vielen PCs, erst noch mit der französischen Tastatur kämpfend, unsere E-Mail lesen.

Das erste Mittagessen definierte ein Niveau, das die Küche der Abbaye nicht mehr unterschritt, sondern immer wieder überbot. An den großen runden Tischen, die mit Klappstühlen »à la camping« umstellt waren, kam jedesmal wieder fröhliche Stimmung auf und der schon mittags gereichte Wein begann immer besser zu munden, je länger das Treffen dauerte. Nach den mehrgängigen Abendessen zog uns unsere Diskussion über wissenschaftstheoretische

Fragen, über Mathematik und Philosophie so in ihren Bann, dass wir meist wieder zu den letzten gehörten, die die Abendrunde verließen. Wieviel Wein wir dabei getrunken haben, darf ich nicht erzählen, das habe ich versprochen.

Unsere neugierige Frage, wie die Abbaye zu diesen großen Tischen gekommen war, beantwortete sich am nächsten Tag, als ich in der Früh einen Arbeiter mit einer riesengroßen Spanplatte auf dem Flur sah, die er wie ein übergroßes Reifrad bzw. wie ein Spielzeug dahinrollte. So passte sich die Tagungsstätte der Zahl der Teilnehmer an, die immer größer wurde, je näher das Galadiner, also der Abend rückte, an dem Donald Knuth und Hermann Zapf sich zu einer Podiumsdiskussion und Ehrung einfinden sollten. Der Abend ist mir nicht nur wegen des beeindruckenden Ambientes des als Festsaal dienenden großen Kirchenschiffes, sondern auch wegen des ausgezeichneten Essens, des guten Weines und des kontrastreichen Auftretens von Donald Knuth und Hermann Zapf in guter Erinnerung.

Hermann Zapf sprühte vor Bühnenfreude und beeindruckte mit seinen Kohleskizzen von Zapfino, Optima und den verschiedenen Varianten, die letztere Schrift erlebt hat. Besonders interessant war, dass man die alte Optima von der neuen dadurch unterscheiden kann, dass – wenn ich mich richtig erinnere – der »Aufstrich« der Ziffer 1 einmal die Form einer Pinocchionase und das andere Mal die eines Entenschnabels hat.

Donald Knuth berichtete – wie mir schien, leicht depressiv verstimmt – über die Frühzeit von T_EX und erntete zustimmendes Gelächter, als er erzählte, wie er die Stadtverwaltung bat, ein häßliches Geschwindigkeitsbeschränkungsschild (was für ein Wort!) auf seine Kosten gegen eines mit ästhetisch geformten Buchstaben auszutauschen. Seiner Bitte wurde nicht entsprochen, und so musste er stets einen Umweg in die Universität in Kauf nehmen, um dem Anblick dieses Schildes zu entgehen.

Wie im Flug ging die Woche vorbei. Als ich am Freitag nach einem vorzüglichen Mittagessen aufbrach, um nach Hause zu fahren, hatte ich nicht nur die beruhigende Gewissheit, dass der Tank meines Autos randvoll war, sondern auch, dass es um die Aussichten, Fichte – oder eine andere geisteswissenschaftliche Ausgabe – mit L^AT_EX zu setzen, viel besser steht als vor 25 Jahren, wenngleich die noch zu leistende Arbeit nicht zu unterschätzen ist.

Was mich aber schon auf dem Heimweg plagte und Selbst immer wieder in den Hintergrund drängte, war die Frage, was soll ich im Tagungsbericht schreiben? Die Vorträge wiederzugeben wäre Unsinn, das könnten Leute, die jahre-

lang an der Universität L^AT_EX benutzten, lehrten und weiterentwickelten, viel besser als ich, der sich »auf der Flucht vor Billy Gates«, mit dem Anspruch, schöne Texte zu schreiben, in den freien Arbeitsstunden mit »trial and error« und vielen Büchern L^AT_EX anzueignen versucht. Die Frage, was schreibe ich, wurde zur Gewissensfrage, aber mir fiel nichts ein. Das spontane Gefühl, ich gehöre doch gar nicht so zur L^AT_EX-Gemeinde wie die Mathematiker und Naturwissenschaftler machte mir zwar klar, dass diese die Mehrheit bilden, aber das führte mich letztlich auch zu keiner Lösung. Also legte ich die Aufgabe erst einmal zur Seite, tröstete mich damit, dass Kreativität nicht zu steuern sei und hoffte, dass mir in der nächsten Zeit noch ein »Licht aufgehen würde«.

Ich genoss die Landschaft und begleitete Selb weiter durch seinen spannenden, verwickelten Kriminalfall und hoffte, dass ich noch vor dem Ende meiner Heimreise erfahren würde, warum das Hörbuch »Selbs Betrug« betitelt war.

Literatur

- [1] Michel Goosens, Mittelbach F. und A. Samarin: *The L^AT_EX Companion*; Addison-Wesley; Boston; 1993.
- [2] David Kastrup: „*De Ore Leonis*“ – *Macroexpansion für Virtuosen*; 2002; Vortrag auf der 26. DANTE-Tagung in Erlangen.
- [3] Donald E. Knuth: *The T_EXbook*; American Mathematical Society; 1984.
- [4] Helmut Kopka: *L^AT_EX*; Addison-Wesley; München; 1988.
- [5] Leslie Lamport: *L^AT_EX: A Document Preparation System*; Addison-Wesley; München; 1994.
- [6] R. Lauth, H. Jacob und H. Gliwitzky (Hg.): *Fichte, Johann G.: Gesamtausgabe der Bayerischen Akademie der Wissenschaften*; Frommann Verlag; Stuttgart; 1962 ff.
- [7] Heiko Oberdiek: *Rund um die geschweifte Klammer*; 2002; Vortrag auf der 26. DANTE-Tagung in Erlangen.
- [8] Joane K. Rowling: *Harry Potter and the Philosopher's Stone*; Bloomsbury; London; 1997.
- [9] Bernhard Schlink: *Selbs Betrug*; Diogenes Verlag; Zürich; 1992.
- [10] Peter K. Schneider und Erich Ruff: *Der begriffene Wahnsinn*; Campus Verlag; Frankfurt; 1985.

Die EuroT_EX – oder wieso der 16-te ein *runder* Geburtstag ist

Peter Seitz

»Let's T_EX Together« – unter diesem Motto stand die 15. europäische T_EX-Tagung EuroT_EX, die vom 7. bis 11. Februar 2005 im Kloster Abbaye des Prémontrés in Pont-à-Mousson (Frankreich), im schönen Moseltal stattfand. In den ersten drei Tagen wurde über die neuesten Entwicklungen im Zusammenhang mit T_EX gesprochen, aber auch insbesondere über Schriften und Fortschritte beim Satz. Ehrengäste waren der Entwickler und Vater des Textsatzsystems T_EX, Donald E. Knuth von der Stanford University und Hermann Zapf, einer der führenden Schriftdesigner der Gegenwart und seinerzeit einer der Berater von Donald E. Knuth beim Entwurf der Computer-Modern-Fonts. Außerdem gab es noch einen runden Geburtstag zu feiern. Die französische T_EX Users Group GUTenberg und die deutschsprachige Anwendervereinigung T_EX e.V. feierten ihr sechzehnjähriges Bestehen. Bei $2^{2^2} = 16$ ist das durchaus eine runde Sache. Grund genug, dass sich auf der Tagung etwa 130 Anwender des Textsatzsystems T_EX eingefunden hatten, um sich über die neuesten Entwicklungen rund um Textsatz und Schriften zu informieren.

Viele Teilnehmer sind bereits am Sonntag Abend angereist und man konnte sich in gelöster Atmosphäre bei einem gemeinsamen Abendessen angeregt unterhalten. Die anfänglichen Schwierigkeiten beim Lesen und Verstehen einer französischen Speisekarte hatten sich spätestens nach einem guten Glas Wein beseitigen lassen, und so wurde bis spät in die Nacht diskutiert. Das Tagungszentrum befand sich direkt im umgebauten Kloster, aber Unterbringung und Verpflegung hatten durchaus nichts von einem »kargen« Klosterleben an sich.

Die Tagung wurde am Montag von Daniel Flipo und Volker RW Schaa eröffnet. Die ersten Vorträge am Vormittag beschäftigten sich schon sehr tief mit komplexen Anwendungen. Gleich der erste Vortrag setzte sich mit den Möglichkeiten auseinander, wie man Dokumente, welche mehrere Sprachen enthalten, setzen kann. Aspekte der Mikrotypographie wurden im zweiten Vortrag behandelt. So wurde erläutert, dass beim Satz von Dokumenten meist viele Informationen über die Buchstaben und Zeichen verloren gehen, etwa Abstände zu anderen Buchstaben. Nach einer erholsamen Kaffeepause wurde es wissenschaftlicher. Der Versuch, eine gemeinsame Terminologie für automatisierte Satzsysteme zu schaffen, wurde vorgestellt. Dies soll es erleichtern,

verschiedene Systeme miteinander zu vergleichen, da genau definiert ist, was mit welchen Begriffen gemeint ist. Der Vormittag konnte dann mit einem Vortrag über das Design einer Computersprache abgeschlossen werden, mit deren Hilfe man einen T_EX-Nachfolger leichter implementieren könnte, ohne dass nur ganz wenige Personen diesen Code verstehen.

Nach dem Mittagessen wurden die Pläne des CTAN-Teams von Jim Hefferon vorgestellt. Diese haben auch Auswirkungen auf das Ansehen von T_EX in der Öffentlichkeit, da ein Anwender relativ schnell mit dem CTAN-Netzwerk konfrontiert wird. Ziel ist es, Informationen auf diesen Servern leichter zu finden und dem Endbenutzer ein besseres Benutzerinterface zur Verfügung zu stellen. Natürlich soll durch diese Maßnahmen die Verbreitung von T_EX besser gefördert werden. Am Nachmittag wurde dann unter anderem ein Praxisbericht über die Anwendung von T_EX im Zusammenhang mit RTF bzw. XML präsentiert. Bestehende Datenbestände aus RTF-Dokumenten mussten zusammengefasst und in einem neuen Dokument gesetzt werden. Die Aufgabe wurde dadurch gelöst, dass die Daten aus den RTF-Dokumenten nach XML konvertiert wurden und aus diesen Daten neue Dokumente entstanden, welche mit Hilfe von T_EX gesetzt wurden. Der letzte Vortrag am ersten Tag befasste sich mit Möglichkeiten, wie man die Ausgaben von T_EX besser nutzen kann um z. B. eine Ausgabe interaktiv zu erzeugen. Den Abschluss des Vortragsprogramms bildeten die Mitgliederversammlungen von GUTenberg und DANTE e.V. bevor es zum Abendessen mit angeregten Diskussionen ging.

Der zweite Tagungstag wurde von Jerzy Ludwиковski eröffnet. Sebastian Rahtz stellte die Möglichkeiten vor, ein Dokument in XML zu codieren und es dann mit der Mächtigkeit von T_EX zu verarbeiten. Die vorgestellten Lösungen sind derzeit keine befriedigende Antwort, aber Frank Mittelbach stellte im nächsten Vortrag Neuigkeiten vom L^AT_EX3-Projekt vor, die dafür vielleicht doch eine Lösung bieten werden. So wurde erläutert, welche Feinarbeiten das L^AT_EX3-Team durchführt, um die verschiedenen Makropakete besser miteinander arbeiten zu lassen, oder, um neue Features, wie etwa Unicode-Fonts, zu implementieren. Wir dürfen gespannt sein, was ein L^AT_EX3 kann, wenn es fertiggestellt sein wird.

Nach so viel Information über L^AT_EX war eine Kaffeepause dringend angesagt. Frisch gestärkt konnte man dann eine Präsentation von Hans Hagen genießen. Hans hat uns am Beispiel des Journals der niederländischen mathematischen Gesellschaft gezeigt, welche Layouts mit ConT_EXt möglich sind.

Man könnte fast meinen, dass dies mit einem einfachen Textsatzsystem wie T_EX nicht möglich ist und dass dazu Layoutprogramme wie Quark Express oder Pagemaker besser geeignet sind. Es ist faszinierend, welche Möglichkeiten ein Layoutdesigner selbst mit einem »einfachen« Textsatzsystem hat, um so ein lebendiges Layout zu schaffen. Nach diesem visuellen Genuss brachte Adam Twardoch uns die Entstehung und Hintergründe von TrueType- und Type1-Fonts näher, und er erläuterte auch die Möglichkeiten von OpenType-Fonts. Durch eine Demonstration der Möglichkeiten von OpenType konnten dem Publikum die Stärken dieses neuen Formats näher gebracht werden, welches gemeinsam von Adobe und Microsoft entwickelt wurde und schon alleine durch die Obsolete-Erklärung des Type1-Font-Formats durch Adobe im Jahr 1999 den zukünftigen Weg von Schriftfonts definiert.

Nach so viel schwieriger Fonts-Kost gab es ein mehrgängiges Menü im Kloster, bei dem natürlich der Wein nicht fehlen durfte. Frisch gestärkt wurde der Nachmittag mit einem Vortrag von Gerd Neugebauer zum Thema Namespaces für ϵ_X T_EX begonnen. Patrick Gundlach präsentierte dem Publikum das ConT_EXt-Dokumentationsprojekt *contextgarden.net*. Ziel ist es dabei, die (fehlende) Dokumentation für ConT_EXt zu ergänzen und zu erweitern, um das System für eine breitere Anwendergruppe attraktiv zu machen.

Hàn Thê Thành, der Entwickler von pdfT_EX, stellte uns zwei neue Erweiterungen vor: optischer Randausgleich und »font expansion«. Hàn erläuterte in seinem Vortrag, dass diese Erweiterungen schon längere Zeit existieren, jedoch nicht so häufig eingesetzt wurden. Der optische Randausgleich macht das Aussehen eines Dokumentes kompakter, indem bestimmte Zeichen über den Rand hinausgeschoben werden. »Font expansion« bedeutet, dass Blocksatz nicht nur durch Variation der Wortabstände, sondern auch durch minimales »Spreizen« der Zeichen erreicht wird, was zu einem ausgeglicheneren Schriftbild führt. Als Name für diesen Algorithmus wurde *hz* gewählt, um damit die Verdienste von *Hermann Zapf* zu würdigen. Nach so viel Mikrotypographie war eine weitere Kaffeepause angesagt, um die neuen Erkenntnisse eifrig miteinander zu diskutieren.

Johannes Küster berichtete über sein »Newmath«-Projekt, dessen Ziel es ist, das zukünftige Encoding für mathematische Fonts in L^AT_EX beispielhaft zu implementieren und entsprechende Werkzeuge für den Entwurf solcher Fonts zu entwickeln. Bogusław Jackowski und Janusz M. Nowacki berichteten von den Problemen, einen LM-Font mit vielen diakritischen Zeichen zu entwickeln, welcher auch speziell in Osteuropa eine weite Verwendung finden soll.

Durch zahlreiche Beispiele wurde das Auditorium auf Feinheiten hingewiesen, welche den meisten Anwendern wohl verborgen blieben, aber nur deshalb, weil sie mit so viel Detailliertheit bearbeitet wurden.

Den Höhepunkt des zweiten Tagungstages bildete wohl die Podiumsdiskussion mit Donald E. Knuth und Hermann Zapf. Den Vorsitz dieses Podiums – übrigens einer der wenigen Vorträge, zu dem kein Abstract im Tagungsband vorlag – hatten Hans Hagen und Volker RW Schaa. Bei der Diskussion standen die klassischen Fontentwürfe von Knuth und Zapf im Vordergrund. So wurde über Hermann Zapfs Schrift *Optima* aus den fünfziger Jahren und die Schriftfamilie *Computer Modern Roman* von DEK gesprochen. Hermann Zapf hat die *Optima* inzwischen überarbeitet und die neuesten Techniken eingesetzt, um den heutigen typographischen Anforderungen gerecht zu werden; das Ergebnis wurde bei Linotype unter dem Namen *Optima nova* veröffentlicht. Die *Computer-Modern*-Schrift wird gerade von einer polnischen Gruppe unter der Leitung von Bogusław Jackowski und Janusz M. Nowacki überarbeitet. Das Ergebnis sind die bereits erwähnten LM-Fonts. Die beiden Autoren nahmen Stellung zu den Veränderungen und Verbesserungen, welche sie in den neuen Schriften gerne sehen möchten. Die stellenweise sehr ins Detail gehende Diskussion machte dem Publikum viele wertvolle Einsichten aus dem Erfahrungsschatz der beiden Experten zugänglich.

Die Schwierigkeiten bei dem Design der Schriftzeichen wurden durch Herman Zapf graphisch am Beispiel der *Optima* demonstriert. Natürlich wurde auch die eine oder andere Geschichte aus dem Wirken der beiden Gäste erzählt. So hat DEK erläutert, wie die dauernde Beschäftigung mit Typographie dazu führte, dass man sie auch in der Umwelt ständig wahrnimmt. So hatte ihn auf der täglichen Fahrt zur Arbeit ein Schild für eine Geschwindigkeitsbegrenzung mit einer schlechten Typographie so sehr geärgert, dass er dafür einen Umweg in Kauf genommen hat, nur um dieses Schild nicht immer wieder sehen zu müssen. Hermann Zapf nahm die Gelegenheit wahr, erstmals in der Öffentlichkeit anzukündigen, dass er momentan seine *Palatino* überarbeitet, um zum einen die Schrift an die Bedürfnisse heutiger Nutzer anzupassen und zum anderen frei von den Beschränkungen früherer Technologien zu sein. Er hat sich auch dahingehend lobend geäußert, dass man sich in keiner anderen Organisation so engagiert und mit viel Liebe zum Detail mit Typographie beschäftigen würde wie bei DANTE e.V. Diese überaus interessante und lebhaft diskussionsrunde konnte nur durch das anstehende Galadiner beendet werden, welches in der Klosterkirche aufgetischt wurde. Man feierte ausgie-

big das lange Leben von T_EX und die Zukunft, sowie die Entwicklungen, die diesem System nach wie vor seine Berechtigung geben.

Nach einer sehr langen Nacht eröffnete Petr Sojka den dritten Tagungstag, und Thomas Feuerstack hatte die schwierige Aufgabe, zu früher Stunde nach einem langen Vorabend sein ProT_EXt vorzustellen. Diese T_EX-Zusammenstellung soll es Anfängern erleichtern, ein T_EX-System auf der Windows-Plattform zu installieren und anzuwenden. Dieser überaus interessante Ansatz ist auch zukünftig auf der T_EX-Live zu finden und wird sicherlich zur Weiterverbreitung von T_EX und Co. beitragen. Nach einem Vortrag von Jean-Michel Hufflen über verbesserte Bibliographiestyles stellte Antoine Lejay eine Methode vor, wie man Massenbewerbungen mittels L^AT_EX generieren kann – eine in Frankreich durchaus übliche Vorgehensweise für Akademiker, um sich an den verschiedenen Universitäten auf eine Stelle zu bewerben. Vor der Kaffeepause erläuterte Frank-René Schaefer seinen Ansatz, wie man mit wenigen und einfachen Mitteln einem Text Formatierungen mitgeben kann, um ein professionelles Layout zu generieren, ohne komplizierte Befehlsketten zu kennen.

Jérôme Laurens stellte seine Überlegungen zu einer T_EX-Wrapper-Struktur für ein grundlegendes T_EX-Dokumentenmodell vor, um den Austausch von T_EX-Dokumenten einfacher zu machen. Er stellte seine iT_EXMac-Umgebung vor, welche eine Voransicht eines gerade in Arbeit befindlichen pdf-Dokumentes auf einer MacOS-X-Plattform ermöglicht. Den Beweis, dass sich T_EX auch bei kommerziellen Projekten nicht verstecken muss, wurde anschließend von Stephan Lehmke vorgelegt. Für den deutschen Lampenhersteller ERCO wurde ein Lampenkatalog mit einem Umfang von ca. 650 Seiten mit Hilfe von L^AT_EX gesetzt. Diese Lösung hat nicht nur zu mehr Effizienz im pdf-Code geführt, was sich in einer deutlichen Reduzierung der Druckdateigröße widerspiegelte, sondern hat auch eine erhebliche Kostenreduzierung gebracht. Zudem hat die Lösung auch Möglichkeiten eröffnet, den Katalog in mehreren Sprachen zu publizieren. Der letzte Vortrag der Tagung wurde von David Kastrup gehalten. So hat er detailliert die Möglichkeiten seines Pakets *bigfoot* erläutert und auf die Schwierigkeiten hingewiesen, welche das System treffen muss, um einen Umbruch in Fußnotensystemen vorzunehmen. Pünktlich zum Mittagessen konnte die 15. EuroT_EX-Tagung geschlossen werden. Am Nachmittag war die Exkursion nach Nantes angesetzt, und an den beiden darauf folgenden Tagen gab es noch zahlreiche Tutorien.

Die überaus interessanten Themen und die nicht allzuweite Anreise aus Deutschland haben diese Konferenz zu einem vollen Erfolg werden lassen. Nicht zuletzt wegen der Ehrengäste hat sich der Besuch voll gelohnt. Die aktuellen Entwicklungen haben gezeigt, dass es sicherlich noch lange ein »T_EX together« geben wird, wie auch immer ein Nachfolger von T_EX heißen wird.

EuroT_EXD.Knuth, J.Knappen,
D.Nickalls, K.Lagally

H.Schlegel, H.Zapf



CTAN – Robin Fairbairns



D.Knuth, K.Höppner, H.Zapf

Geburtstagstorte mit
Zapf-Schrift

Fotos: Bernd Raichle

Bretter, die die Welt bedeuten

Neues von $\varepsilon_{\chi}\text{T}_{\text{E}}\text{X}$: TFM-Daten anschaulich darstellen

Michael Niedermair

Dieser Artikel beschreibt, wie mit Hilfe von $\varepsilon_{\chi}\text{T}_{\text{E}}\text{X}$ eine TFM-Datei durch eine Transformation in eine gestaltete HTML-Datei umgewandelt werden kann, um sie optisch »aufgepeppt« darzustellen.

Einleitung

Mit diesem Artikel möchte ich eine Serie einleiten, um in unregelmäßigen Abständen Neues von $\varepsilon_{\chi}\text{T}_{\text{E}}\text{X}$ (siehe <http://www.extex.org>) zu berichten. Als Einstiegsthema möchte ich hier eine etwas andere Darstellung der TFM-Datei ($\text{T}_{\text{E}}\text{X}$ -Font-Metrik) zeigen.

TFM für die Metriken

Ein Ziel von $\varepsilon_{\chi}\text{T}_{\text{E}}\text{X}$ ist es, die traditionellen Fontmetriken von $\text{T}_{\text{E}}\text{X}$ zu verwenden. Dazu müssen sie eingelesen und in ein eigenes Format umgewandelt werden (mit AFM- oder TTF-Dateien wird genauso verfahren). Für die Darstellung der Metriken verwendet $\varepsilon_{\chi}\text{T}_{\text{E}}\text{X}$ ein XML-Format, welches die Dateiendung EFM ($\varepsilon_{\chi}\text{T}_{\text{E}}\text{X}$ -Font-Metrik) besitzt. Zum Testen, ob die Werte richtig ausgelesen wurden, wurden sie in eine XML-Datei geschrieben und dann mit Hilfe eines JUnit-Tests verglichen. Ein JUnit-Test ist ein Test, der u. a. Java-Klassen testet und sehr einfach die Möglichkeit bietet, Werte zu vergleichen (z. B. `assertEquals("-1026142560", el.getAttributeValue("checksum"));`). Das XML-Format hat in dieser Sicht mehrere Vorteile: Es ist direkt für den Menschen lesbar, es lässt sich beliebig erweitern und mit vielen Hilfsprogrammen verarbeiten.

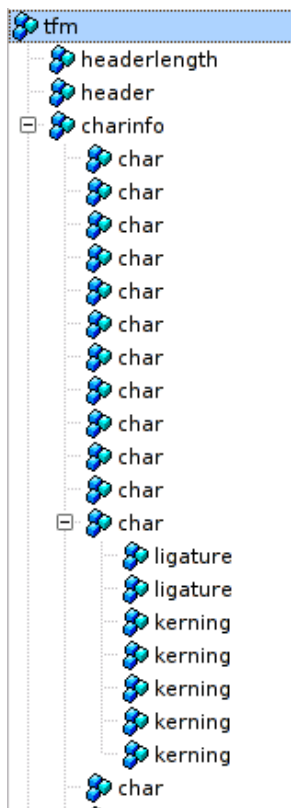


Abbildung 1: Grafische Struktur einer EFM-Datei

Die XML-Datei wird dabei mit $\epsilon\chi\mathrm{T}_{\mathrm{E}}\mathrm{X}$ wie folgt erzeugt (eine entsprechende Installation von $\epsilon\chi\mathrm{T}_{\mathrm{E}}\mathrm{X}$ und Java vorausgesetzt):

```
1 java de.dante.util.font.TFM2XML cmr12.tfm /tmp/cmr12.xml
```

Als Ergebnis erhält man eine XML-Datei (siehe Abbildung 1). Dabei wird jede interne TFM-Tabelle (Informationen zu den Zeichen, Ligaturen, Maßen, zu Kerning usw.) entsprechend umgesetzt. Zusätzlich wird unter `/tfm/charinfo/char` zu jedem Zeichen eine informative Zusammenfassung aus den anderen Tabellen des Zeichens bereitgestellt. Im Gegensatz zur XML-Datei



Abbildung 2: Ansicht mit dem Browser

enthält die EFM-Datei nur die zusammengesetzten Informationen und nicht mehr alle Originaltabellen. Zum Testen werden aber alle Informationen benötigt. Tabelle 1 zeigt eine Übersicht über die Java-Klassen, die TFM-Dateien umwandeln.

Mit XSL-T transformieren

Eine Möglichkeit, diese XML-Datei weiterzuverarbeiten, stellt XSLT dar. (XSL-Transformation ist eine Möglichkeit, mit Regeln eine XML-Datei in verschiedene andere Formate, wie z. B. XML, HTML, $\text{\LaTeX}$ usw., umzuwandeln. Dabei können entweder die ganze XML-Datei oder nur Teilbereiche davon für die Ausgabe verwendet werden.) Für unser Beispiel werden nur einige Informationen aus der XML-Datei transformiert und als Ausgabeformat HTML festgelegt. Der Aufruf von XSLT lässt sich auch direkt mit $\epsilon\chi\text{T}_{\text{E}}\text{X}$ durchführen, da Java einen XSLT-Prozessor integriert hat. Ein Beispiel einer solchen XSLT-Datei ist in $\epsilon\chi\text{T}_{\text{E}}\text{X}$ enthalten.

```
1 java de.dante.util.xslt.Transform /tmp/cmr12.xml
2   src/xslt/tfm2html.xsl /tmp/cmr12.html
```

The screenshot shows a web interface for viewing font metrics. At the top, the character 'f' is displayed. Below it, a table lists various metrics for this character. The table is organized into two main sections: general metrics and detailed character metrics.

width	height	depth	italic
299.184702	924.443702	0	45.73362

letter-id	letter	lg-id	lg
105	f	12	
100	f	13	
108	f	13	

id	char	size
89	f	45.73362
82	f	45.73362
33	f	45.73362
41	f	45.73362
92	f	45.73362

Abbildung 3: Ansicht des Zeichens »f«

Verwendet man zusammen mit der HTML-Datei eine CSS-Datei – ein Beispiel ist bei $\epsilon\chi\text{T}_{\text{E}}\text{X}$ enthalten –, so kann man das Aussehen deutlich aufpeppen (siehe Abbildung 2). Dabei werden zuerst allgemeine Informationen des Fonts und eine Übersicht (mit Link) aller vorhandenen Zeichen angezeigt. Durch einen Klick kommt man dann direkt zu den Informationen des entsprechenden Zeichens (siehe Abbildung 3). Dabei werden die Breite, Höhe, Tiefe und der Wert für italic-Korrektur angezeigt. Ist für den Buchstaben eine Ligatur definiert, wird diese direkt darunter angezeigt, ebenso wird mit dem Kerning verfahren.

Ansicht anpassen

Wem die Ansicht oder die Darstellung der Informationen nicht gefällt, kann sehr leicht eine entsprechende Anpassung durchführen. $\epsilon\chi\text{T}_{\text{E}}\text{X}$ mit allen seinen Dateien findet man unter <http://www.extex.org>.

Zusammenfassung

Mit $\epsilon\chi\text{T}_{\text{E}}\text{X}$ lassen sich TFM-Dateien in XML, EFM oder in andere Formate wie PL einfach umwandeln. Durch das flexible XML-Format lässt sich aus diesen Informationen eine beliebige andere Darstellung erzeugen. $\epsilon\chi\text{T}_{\text{E}}\text{X}$ verwendet übrigens XML-Dateien nicht nur für die Metriken, sondern für vieles mehr. Es gibt sogar schon ein XML-Backend (Testphase), der alle Informa-

Java-Klasse	Beschreibung
TFM2XML	Wandelt eine TFM-Datei in eine XML-Datei um (alle Informationen der TFM-Datei werden komplett übernommen).
TFM2EFM	Wandelt eine TFM-Datei in eine EFM-Datei um. Es werden nur die zusammengesetzten Informationen verwendet, die der Typesetter benötigt.
TFM2PL	Wandelt eine TFM-Datei in das PL-Format (wie <i>tftopl</i>) um. (Auch hier vergleicht ein JUnit-Test die beiden Ergebnisse und meldet entsprechende Fehler).

Tabelle 1: Java-Klassen, die TFM-Dateien umwandeln

tionen des Typesetters in eine XML-Datei schreibt, so dass man mit einer entsprechenden Transformation beliebige andere Formate erzeugen kann.

Schach mit *skak.sty*

Ulrike Fischer

Vor etwa 10 Jahren gab es den letzten Artikel zum Thema Schach in »Die $\text{\TeX}$ nische Komödie«. Damals wurde das Paket **chess** vorgestellt. Im folgenden Artikel wird der Nachfolger **skak.sty** vorgestellt. Die wesentlichen Änderungen betreffen die Eingabe: **skak.sty** versteht die Standardschachnotationen FEN und PGN. Es gibt aber noch mehr zu entdecken.

Installation

Beschrieben wird die Version 1.4 Beta 6 [1]. Ich empfehle dringend, diese Version zu installieren, da sie diverse Bugs korrigiert. Die *.zip*-Datei ist etwas unordentlich, aber lassen Sie sich davon nicht abschrecken. Eigentlich brauchen Sie nur *skak.sty*, *lambda.sty* und die diversen *.mf*-Dateien für die Schriften. Sie sollten alte *.tfm*- und *.pk*-Dateien der Schriften in Ihrem System löschen, da auch in den Schriften Bugs korrigiert wurden. Benötigt wird außerdem das Paket **chessfss.sty**, das es auf CTAN

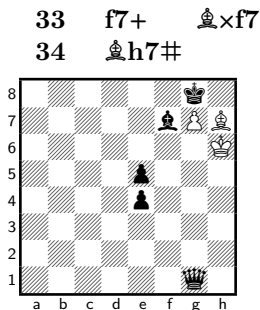
gibt [2]. Sinnvoll ist die Installation von *skaknew* [3], der Type1-Version der Schachschriften. Mit der neuen *skak.sty*-Version lassen sie sich einfach mit `\usepackage[skaknew]{skak}` benutzen. Die Dokumentation von *chessfss.sty* enthält dazu weitere Hinweise.

Einführung

Abbildung 1 zeigt eine einfache Schachpartie. Zuerst wird mit dem Befehl `\fenboard{<FEN>}` die Anfangsstellung angegeben. Anschließend werden mit `\mainline{<PGN-Partie>}` ein paar Züge eingegeben. Diese werden formatiert und gedruckt. Außerdem erkennt *skak.sty* die Züge, aktualisiert das interne Brett, das anschließend mit `\showboard` gedruckt werden kann. Diese drei Phasen – Stellungen initialisieren, Züge eingeben und drucken, Diagramme ausdrucken – werde ich nun etwas genauer beschreiben.

Eine Stellung initialisieren

Eine Schachstellung kann man mit `\fenboard{<FEN>}` angeben, wobei FEN die Forsyth-Edwards-Notation [4] der Stellung ist. Die meisten Schachprogramme können Partiestellungen in FEN ausgeben. Für die Anfangsstellung (FEN: `rnbqkbnr/pppppppp/8/8/8/8/PPPPPPPP/RNBQKBNR w KQkq - 0 1`) gibt es die Abkürzung `\newgame`. Die Stellung muss keine legale Schachstellung sein. Auch ohne Türme dürfen Sie behaupten, dass alle Spieler noch



```

1 \styleC% Züge in Tabellenform
2 \fenboard% Anfangsstellung eingeben
3 {6k1/6P1/5PBK/3bp3/4p3/8/8/6q1 w - - 0 33}
4 \mainline{% ein paar Züge machen
5 33. f7+ Bxf7 34. Bh7#}
6
7 \showboard% Diagramm drucken
8 \captionof*{figure}{Stellung nach
  \styleUF\lastmove.}

```

Stellung nach 34. ♔h7#.

Abbildung 1: Ein einfaches Beispiel

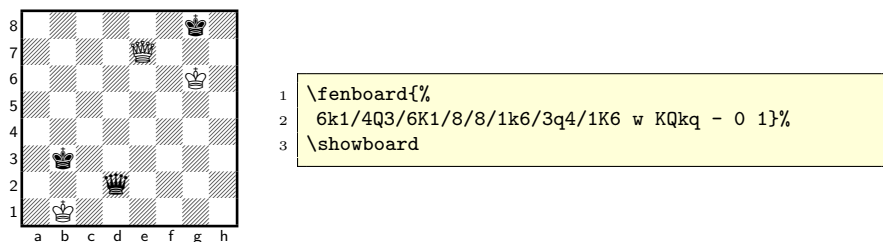


Abbildung 2: Zwei Aufgaben auf einem Brett

rochieren können. Es macht nichts aus, wenn Weiß am Zug, aber Schwarz im Schach steht, oder mehr als zwei Könige auf dem Brett sind (siehe Abbildung 2). Sie sollten aber derartige illegale Stellungen nicht als Startpunkt für einen `\mainline`-Befehl nehmen. Der Befehl `\boardasfen` druckt die FE-Notation der aktuellen Stellung des internen Brettes.

Abspeichern von Stellungen

`\storegame{<label>}` speichert die aktuelle Stellung in einem internen Befehl. Mit `\restoregame{<label>}` können Sie diese Stellung dann an anderer Stelle wieder aktivieren. `\savegame{<partie>}` speichert die aktuelle Stellung in eine Datei `{<partie>}.fen`, die dann mit `\loadgame{<partie>}` geladen und aktiviert werden kann.

Der Diagrammdruck

Ein Diagramm der aktuellen Stellung erhält man mit `\showboard`. Dabei ist Weiß unten. Mit `\showinverseboard` erhält man ein Brett mit Schwarz unten.

Die Brettgröße

Mit dem Befehl `\setupboard{<Länge>}{<Länge>}` kann man die Brettgröße definieren. Das erste Argument gibt die Breite eines Feldes an, das zweite die Größe der Brettbeschriftung.

Einige Brettgrößen sind vordefiniert und können über einen Befehl oder – für das ganze Dokument – über eine Paketoption aufgerufen werden. Vor-

eingestellt ist `\normalboard`. Die Linie um das Brett lässt sich derzeit nicht verändern, sie ist immer ein 1pt breit. Wenn Ihnen das zu dick ist, können Sie hilfsweise große Bretter verwenden, die Sie mit `\resizebox` aus dem `graphicx`-Paket verkleinern.

Befehl	Paketooption	entspricht
<code>\tinyboard</code>	<code>tiny</code>	<code>\setupboard{10pt}{6pt}</code>
<code>\smallboard</code>	<code>small</code>	<code>\setupboard{15pt}{8pt}</code>
<code>\normalboard</code>	<code>normal</code>	<code>\setupboard{20pt}{10pt}</code>
<code>\largeboard</code>	<code>large</code>	<code>\setupboard{30pt}{12pt}</code>

Die Beschriftung

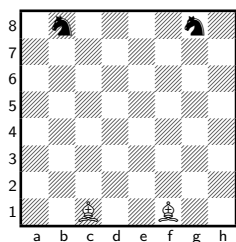
Die Beschriftung des Brettes kann man mit `\notationOff` und `\notationOn` oder über die Paketooptionen `notation` und `notationoff` an- und abstellen (Voreinstellung ist »an«). Zu beachten ist jedoch, dass beide Befehle die Definitionen von `\showboard` und `\showboardinverse` überschreiben. Man sollte sie daher nicht einsetzen, wenn man die Brettbefehle »gepatcht« hat.

Nach `\showmoverOn` oder wenn man *skak.sty* mit der Paketooption `mover` geladen hat, wird rechts vom Brett angezeigt, wer am Zug ist. Der Befehl benutzt `pstricks`-Makros, er funktioniert nur zusammen mit der Paketooption `ps`. Möchte man pdfL^AT_EX verwenden, muss man wie beim Highlighting mit z. B. `pst-pdf` arbeiten. Mit `\showmoverOff` kann man die Anzeige wieder abstellen.

Die Zahlen links und der Zuganzeiger ändern die Breite des Brettes nicht, sie ragen aus der Box heraus. Möchte man z. B. mehrere Bretter nebeneinander setzen, muss man selbst für geeigneten Abstand sorgen. Die Buchstaben unten ändern hingegen die Grundlinie des Brettes.

Figuren ausblenden

Mit dem Befehl `\showonly{<Figuren>}` kann man Figuren auf dem Brett ausblenden (siehe Abbildung 3). Das Argument ist eine durch Kommas getrennte Liste der englischen Bezeichnung der Figuren (K, Q, R, B, N, P, k, q, r, b, n, p). Weiße Figuren werden mit Großbuchstaben, schwarze mit Kleinbuchstaben angegeben. Die Wirkung der Befehle `\showall`, `\showallbut{<Figuren>}`, `\showonlypawns`, `\showonlyblack` und `\showonlywhite` sollte klar sein.

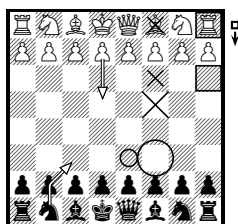


```

1 \newgame
2 \showonly{B,n}% nur weiße Läufer und schwarze
   Springer zeigen
3 \showboard

```

Abbildung 3: Figuren ausblenden



```

1 \newgame\showmoverOn\notationOff
2 \showinverseboard
3 \highlight{a1,a3}
4 \highlight[x]{c1,c3}
5 \highlight[X]{c4}
6 \highlight[o]{d6}
7 \highlight[O]{c6}
8 \printarrow{e2}{e4}
9 \printknightmove{g8}{f6}

```

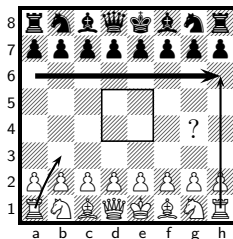
Abbildung 4: Verschiedene Markierungen

Markieren von Feldern und Zügen

skak.sty bietet diverse Befehle zum Markieren des Brettes an. Abbildung 4 zeigt verschiedene Möglichkeiten. Damit man sie verwenden kann, muss man **skak.sty** mit der Paketooption **ps** laden. Die Befehle benutzen **pstricks** und andere Postscriptbefehle, funktionieren also nicht ohne weiteres mit pdf- $\text{\LaTeX}$.

Möchte man den Stil der Pfeile ändern, muss man den internen **psstyle** ändern. Am sinnvollsten kopiert man sich **\newpsstyle{pskak}{...}** aus **skak.sty** in die eigene Präambel und ändert dort die Werte.

Mehr Gestaltungsfreiheit erhält man, wenn man auf einige der internen Befehle zurückgreift: **skak.sty** setzt bei jedem **\showboard**-Befehl einen Knoten (»Node«) mit dem Namen »BM« mitten auf das Feld rechts unten (h1 oder a8). Mit dem Befehl **\set@special@psmarkers{<Feld1>}{<Feld2>}** werden zwei Knoten »From« und »To« gesetzt. Diese drei Knoten können von



```

1 \newgame
2 \showboard
3 \makeatletter
4 \set@special@psmarkers{a1}{b3}
5 \ncarc{->}{From}{To}
6 \set@special@psmarkers{a6}{h6}
7 \ncline[linewidth=0.2]{->}{From}{To}
8 \ncline{->}{BM}{To}
9 \pscustom[linewidth=.06]{%
10   \skak@translate d4 \psframe(-.5,-.5)(1.5,1.5)}
11 \set@special@psmarkers{g4}{g4}
12 \ncline{-}{From}{To}\mput*{?}
13 \makeatother

```

Abbildung 5: Mehr Markierungen

`pstricks`-Befehlen genutzt werden. Der Befehl `\skak@translate` *Feld* (ohne Klammern) verschiebt den Ursprung des Koordinatensystems in das angegebene Feld. Er kann z. B. in `\pscustom`-Befehlen benutzt werden. Abbildung 5 zeigt einige Beispiele.

Highlighting mit pdf $\LaTeX$

Mit dem Paket `pst-pdf` [6] ist es möglich, die Befehle, die auf `pstricks` beruhen, mit pdf $\LaTeX$ zu verwenden. Man muss nur darauf achten, dass die Bretter in einer genügend großen Box stecken, die die Notation, den Zuganzeiger und alle Kurven und Pfeile umfasst. Die Abbildungen 4, 5 und 6 wurden mit Hilfe von `pst-pdf` erstellt.

```

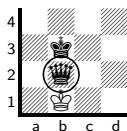
1 \begin{postscript}[trim = \squarelength{} Opt \squarelength{} Opt]
2   [ ... ]
3 \end{postscript}

```

Weitere Informationen findet man in der Paketdokumentation.[6]

Brettausschnitte

In Schachbüchern werden oft nur halbe oder viertel Bretter verwendet, um Platz zu sparen. `skak.sty` hat dazu keine geeigneten Befehle. Es wäre nicht einfach, solche Befehle hinzuzufügen: Man müsste viele interne Befehle neu



```

1 \SpecialCoor \psset{unit=\squarelength}
2 \newcommand\boardrulecorr{1pt}
3 \fenboard{8/8/8/8/1k6/1q6/1K6 w - - 0 1}
4 \begin{pspicture*}(-1,-1)(4,4)
5 \pnode(8,8){A}
6 \rput[rt]([nodesep=\boardrulecorr,
7           offset=\boardrulecorr]A){\showboard}
8 \highlight[0]{b2}
9 \end{pspicture*}

```

Abbildung 6: Brett Ausschnitte

definieren und viele Varianten berücksichtigen. Glücklicherweise gibt es aber **pstricks**. Abbildung 6 zeigt eine von mehreren möglichen Varianten, das Brett zu beschneiden. Man darf beim Schneiden einige Dinge nicht vergessen: Erstens ist `\squarelength` die Breite eines Feldes, zweitens hebt die Notation das Brett um einen unbekannten Abstand, man sollte daher besser eine der oberen Ecken als Bezugspunkt wählen, und drittens sollte man nicht vergessen, dass der Rahmen 1pt breit ist.

Züge eingeben

skak.sty stellt mehrere Befehle zur Verfügung, die Notationen von Zügen verarbeiten. Einige Befehle erkennen die Züge und aktualisieren das interne Brett, einige drucken nur die Züge mehr oder weniger gut formatiert, einige tun beides. Tabelle 1 gibt einen groben Überblick, was die einzelnen Befehle können und tun. Alle Befehle verarbeiten Partie- und Zugnotationen in PGN (Portable Game Notation [5]). Die Möglichkeit, Partien in PGN einzugeben, stellt eine große Verbesserung gegenüber dem alten Paket **chess** dar. Partien können nun mit Hilfe eines Schachprogramms mit der Maus eingegeben werden. Das fehlerträchtige und aufwändige Eintippen von beispielsweise `\ply e2e4` und `\move c8c4 e7d8` gehört der Vergangenheit an. Daher kann man nun auch umfangreiche Bulletins schön formatiert und mit Bildern aller Endstellungen erstellen.

Diese Erleichterung bekommt man aber nicht ganz umsonst. PGN ist für Menschen sehr leicht zu lesen und zu verstehen, für einen Computer gibt es aber diverse Fallstricke. Der Parsingalgorithmus von **skak.sty** ist daher relativ langsam, er kommt nur mit einfachen PGN-Partien zurecht und mag

Befehl	Drucken	Formatierung	Parsen
<code>\movecomment</code>	✓	–	–
<code>\wmove</code>	✓	–	–
<code>\bmove</code>	✓	–	–
<code>\lastmove</code>	✓	✓	–
<code>\variation</code>	✓	✓	(teilweise)
<code>\mainline</code>	✓	✓	✓
<code>\hidemoves</code>	–	–	✓

Tabelle 1: Überblick über die Befehle zum Eingeben und Drucken von Zügen

Fehleingaben gar nicht. 0-0 statt 0-O, fehlende Leerzeichen, falsche Zugnummer und falsche Züge sind die häufigsten Fehlerquellen. Und man sollte immer damit rechnen, dass irgendwo sich noch ein Bug versteckt.

skak.sty enthält Befehle, mit denen man die englischen Bezeichnungen der Figuren bei der *Eingabe* auf andere Sprachen umstellen kann. Ich halte dies für ziemlich nutzlos: Partien sollte man sowieso nicht von irgendwelchen Formularen abtippen. Meiner Erfahrung nach enthalten 95% der Notationen bei Turnieren mindestens einen Fehler. Mit einem Schachprogramm ist es viel einfacher und das kann alle PGN mit den englischen Bezeichnungen ausgeben.

Die sensiblen Stars: `\hidemoves`, `\mainline` und `\variation`

Bei diesen drei Befehlen *muss* das Argument $\{\langle PGN \rangle\}$ mit einer Zugnummer beginnen (einer Zahl mit einem Punkt falls ein weißer Zug folgt oder einer Zahl mit drei Punkten falls ein schwarzer Zug folgt). Danach kommen die Züge getrennt durch Leerzeichen.

Bei `\hidemoves` und `\mainline` *müssen* außerdem die Zugnummer und die Zugfarbe zur aktuellen internen Stellung passen und es dürfen keine unmöglichen Züge gemacht werden. (Dies ist nicht ganz richtig; *skak.sty* überprüft nur teilweise, ob ein Zug legal ist. Es ist z.B. möglich aus dem Schach zu rochieren oder eine gefesselte Figur zu ziehen. Nicht möglich ist es jedoch, z. B. mit dem Turm diagonal zu ziehen.) Man kann die Partie jederzeit unterbrechen und später mit einem neuen `\mainline` oder `\hidemoves` wieder aufnehmen.

1. e4 *wieso dies?* e5!?

2. ♖f3±↑ ♘c6

3. ♙c4 ZZ⊂Bc5

```

1 \newcommand\Kom{{\normalfont\itshape~wieso dies?}}
2 \newgame
3 \mainline{
4   1. e4\Kom{ } e5!? 2. Nf3\wbetter\within{ } Nc6} %
   das geht

```

```

1 \mainline{3. Bc4 \betteris Bc5} %das nicht

```

Abbildung 7: Kommentare

`\variation` ist nicht ganz so empfindlich. Der Befehl versucht zwar ebenfalls die Züge zu erkennen, aktualisiert aber nicht die interne Stellung. Er akzeptiert in der Eingabe daher auch unmögliche Züge. Das Argument muss aber mit einer Zugnummer beginnen und formal einer Abfolge von Zugnummern, weißen und schwarzen Zügen gleichen. (Hier hat es von Version 1.3 zu 1.4 von **skak.sty** eine Neuerung gegeben. Früher durchlief `\variation` gar nicht den Parsingalgorithmus.)

skak.sty versteht nur einfache Partiennotationen. Folgende Elemente von PGN werden (derzeit) nicht unterstützt:

1. Die Notation darf keine NAG-Kommentare (Numeric Annotation Glyphs) enthalten. Dabei handelt es sich um Zahlen mit einem Dollarzeichen davor, die für bestimmte Kurzkomentare stehen (\$13 z. B. steht für `\unclear`: ∞). **skak.sty** stellt eine Reihe von Befehlen bereit, die man als Ersatz für die NAG-Kommentare verwenden kann. *Nach* einem Zug können Sie die Befehle problemlos verwenden; man muss nur darauf achten, den Befehl mit `{ }` zu beenden, damit das trennende Leerzeichen nicht verschluckt wird. *Vor* einem Zug bringen sie den Parsingalgorithmus durcheinander wie Abbildung 7 zeigt. Mit der neuen Version 1.4 werden die Symbole nicht mehr von **skak.sty** sondern vom Paket **chessfss.sty** definiert. In dessen Dokumentation finden Sie auch eine Liste zu diesen Symbolen.
2. Die Notation darf keine Textkommentare enthalten (PGN benutzt zur Kennzeichnung solcher Kommentare geschweifte Klammern `{ }`). Abbildung 7 zeigt eine Möglichkeit, diese Beschränkung zu umgehen. Das doppelte Klammerpaar ist kein Versehen, es schützt den Inhalt vor dem Parsingalgorithmus!

3. Die Notation darf keine Varianten enthalten (PGN benutzt zur Kennzeichnung von Varianten eckige Klammern []). Varianten muss man entweder durch geeignete `\mainline/\variation`-Befehle ersetzen oder man verwendet das neue Paket `texmate.sty`.

Exkurs: PGN-Dateien für *skak.sty* vorbereiten

PGN-Dateien enthalten nicht nur Partienotationen, sondern auch in PGN-Tags Informationen zu den Spielern, der Eröffnung, dem Turnier u. a. PGN-Tags sind durch eckige Klammern gekennzeichnet und stehen vor der Notation der Partie. Für ein Bulletin sollte man sinnvollerweise die Tags und die Partiennotationen *skak.sty*-gerecht vorbereiten. Dazu sind mir zwei Möglichkeiten bekannt:

Zum einen kann man mit dem Programm `pgn2ltx` [7] PGN-Dateien säubern. Es fügt um die Partien geeignete `\mainline`- und `\newgame`-Befehle ein, löscht störende Kommentare, trennt die Hauptzüge von den Varianten, ersetzt die NAGs und bearbeitet auch die PGN-Tags: [`White "Spieler"`] wird zu `\def\WhitePlayer{Spieler}`. Es hat allerdings einen kleinen Bug: Es löscht das Gleichheitszeichen aus einer Bauernumwandlung und bringt damit *skak.sty* aus dem Tritt. Außerdem benützt es für Varianten den `\movecomment`-Befehl statt des besseren `\variation`-Befehls.

Zum anderen gibt es ein Perl-Modul `Chess::PGN::Parse` [8], mit dem man sich eigene Umwandlungsskripte schreiben kann. Das macht natürlich mehr Arbeit, dafür hat man mehr Einfluss auf das Ergebnis.

Die Robusten: einfache Druckbefehle

Die Befehle `\movecomment`, `\wmove`, `\bmove` können zum Drucken einzelner Züge verwendet werden. Sie fügen Formatierungsbefehle nur am Anfang ein und ersetzen ansonsten nur mehr oder minder stupide die schachspezifischen Zeichen:

- Die Buchstaben `KQRBNp` werden zu ♔♚♙♜♞♝♞♟,
- `+` wird zu `+` (`\checksymbol`),
- `0-01` wird zu `O-O` (`\castlingchar\castlinghyphen\castlingchar`)²,

¹`0-0`, nicht `0-0!`

²Standardmäßig wird in *skak.sty* ein großes O für die Rochade verwendet, ich mag 0 lieber, daher habe ich mir `\castlingchar` umdefiniert.

- $\times$ wird zu $\times$ (`\capturesymbol`),
- $\#^3$ wird zu $\#$ (`\mate`) und Gleichheitszeichen $=$ werden gelöscht: Beispielsweise wird $e8=Q$ zu $e8\text{♖}$.

Das Ersetzen der Zeichen stellt keine großen Anforderungen an $\text{T}_{\text{E}}\text{X}$. Die Befehle `\movecomment`, `\wmove` und `\bmove` akzeptieren daher so ziemlich jedes Argument, solange kein Absatzende enthalten ist.

$\text{♜f3}, \dots \text{♜f6},$	1	<code>\wmove{Nf3}, \bmove{Nf6}, \\\</code>
$\text{♖uatsch}, \text{♚rei und } \times$	2	<code>\movecomment{Quatsch, Brei und x}</code>

Exkurs: Varianten mit *texmate.sty*

texmate.sty ist ein neues Paket zum Drucken von Schachpartien. Es parst die Eingabe nicht, erkennt also keine Züge. Es hat auch noch einige Kinderkrankheiten (z. B. erkennt es nicht die PGN-Standards O-O für die Rochade und $e8=Q$ für die Bauernumwandlung). Dafür kann es gut mit Varianten umgehen, da es die PGN-Klammern erkennt und vier unterschiedliche Variantenformatierungen unterstützt. Es ist daher einfach, je nach Variantentiefe andere Schriftattribute zu verwenden. *texmate.sty* verwendet allerdings derzeit nur die alten *skak*-Metafontschriften. Ein Wechsel zu anderen Schriftfamilien ist noch nicht möglich.

Die Zusammenarbeit mit *skak.sty* ist noch etwas mühsam: Beide Pakete definieren einen `\newgame`-Befehl, man muss daher mit geeigneten `\let`-Befehlen zumindest einem der Befehle einen neuen Namen geben. Stellungen werden in *texmate.sty* anders initialisiert. *texmate.sty* hat eigene Methoden, die aktuelle Zugnummer und Zugfarbe zu erkennen; kombiniert man `\mainline` mit *texmate.sty*-Varianten, muss man gelegentlich *texmate.sty* auf die Sprünge helfen und korrigieren. Außerdem muss man den Stil der Notation (z. B. die Zeichen und Abstände nach der Zugnummer und zwischen den Zügen) zweimal definieren, weil beide Pakete dafür eigene Befehle haben. Insgesamt aber ist das Paket sehr vielversprechend und kann eine gute Ergänzung für *skak.sty* werden.

³ $\#$ kann Ärger machen, wenn man `\mainline` innerhalb eines anderen Befehls aufruft; im Allgemeinen ist es daher sinnvoll, bereits im PGN $\#$ durch `\mate` zu ersetzen.

Der letzte Zug

Neu in *skak.sty* ist der Befehl `\lastmove`. Er gibt den letzten Zug des letzten `\mainline`-Befehls aus, und zwar inklusive Zugnummer und der im aktuellen Stil definierten Formatierungsbefehle, die zwischen der Nummer und dem Zug eingefügt werden (siehe Abbildung 10). Man muss darauf achten, dass nicht ein Stil (z. B. `\styleC`) aktiv ist, der Befehle enthält, die im Text nicht erlaubt sind.

Die Formatierung des Ausdruckes: Stile

Die einfachen Druckbefehle fügen Formatierungsbefehle nur zu Beginn und am Ende ein. Die strengerer Anforderungen an die Syntax ermöglichen es den anspruchsvolleren Befehlen `\mainline` und `\variation` ihr Argument in die Teile »Zugnummer«, »weißer Zug« und »schwarzer Zug« zu zerlegen und zwischen diesen Teilen diverse Befehle zur Formatierung einzufügen. Man kann daher Züge durch ein Komma trennen, die Zugnummer mit oder ohne Punkt ausgeben, tabbing-Befehle einfügen und vieles Weitere.

Abbildung 9 zeigt alle Formatierungsbefehle, die vor und zwischen den Zügen von den jeweiligen Druckbefehlen eingefügt werden.

Mit den Befehlen `\mainlinestyle` und `\variationstyle` kann man zwischen den Haupt- und Nebenvarianten z. B. die Schrift wechseln. Unterschiedliche Schriften je nach Variantentiefe sind nicht (automatisch) möglich, da empfiehlt es sich, auf *texmate.sty* auszuweichen.

Die Befehle `\opencommands` und `\closecommands` kann man benutzen, um z. B. eine tabbing-Umgebung zu öffnen bzw. zu schließen.

skak.sty enthält drei vordefinierte Formatierungsstile, die über die Befehle `\styleA`, `\styleB` und `\styleC` aufgerufen werden können (siehe Abbildung 8, wobei die Tabellenbreite bei `\styleC` etwas verkleinert wurde). Die drei vordefinierten Stile werden in *skak.sty* mit einem zweistufigen Verfahren definiert: Zuerst werden zu jedem Formatierungsbefehl analoge `\styleX@. . .`-Befehle definiert. In der Definition von `\styleX` selbst wird dann z. B. `\let\afterblack=\styleX@afterblack` gesetzt.

Möchte man einen Stil ändern, ist es daher nicht sinnvoll, Befehle wie `\afterblack` umzudefinieren. Ein Wechsel zu einem anderen Stil würde die neue Definition zerstören. Man muss – wie Abbildung 10 zeigt – die internen Befehle ändern. Es gibt auch kein vernünftiges »Userinterface«, um neue Stile

1. e4, e5 2. ♖f3, ♘c6 2. -, ♘f6

1 e4 e5 2 ♖f3 ♘c6 2... ♘f6

1 e4 e5
2 ♖f3 ♘c6
2... ♘f6

```
1 \styleA\newgame
2 \mainline{1. e4 e5 2. Nf3 Nc6}
3 \variation{2... Nf6}
```

```
1 \styleB\newgame
2 \mainline{1. e4 e5 2. Nf3 Nc6}
3 \variation{2... Nf6}
```

```
1 \styleC\newgame
2 \mainline{1. e4 e5 2. Nf3 Nc6}
3 \variation{2... Nf6}
```

Abbildung 8: Demonstration der drei vordefinierten Stile

zu definieren. Es ist aber relativ einfach, sich einen komplett eigenen Stil zu definieren: Kopieren Sie einfach eine der Definitionen aus `skak.sty`, und ändern Sie Namen und Befehle entsprechend (`\makeatletter`/`\makeatother` nicht vergessen!). Eine gewisse Vorsicht ist dabei angebracht, besonders wenn Sie innerhalb eines Stils zu einem anderen wechseln (wie es `\styleC` bei Varianten tut), oder wenn Sie zu Beginn mit `\opencommands` eine Gruppe oder Umgebung öffnen. Man kann leicht seltsame Nebenwirkungen erzeugen. Der Code von `skak.sty` zeigt, dass auch der Autor immer mal mit solchen Nebenwirkungen gekämpft hat.

Dieses Dokument verwendet einen eigenen lokalen Stil `\styleUF`. Abbildung 9 wurde mit einem lokalen Stil `\styleTest` realisiert. Dessen Stildefinition enthält u. a. die Zeilen

```
1 \def\styleTest@beforenumber
2   {\ttfamily\textbackslash beforenumber}}
```

Die Formatierung des Ausdruckes: Schachschriften

Von der Version 1.4 an ist es möglich, in `skak.sty` andere Schachschriften als die Metafontschrift `skak` zu verwenden. `skak.sty` verwendet nun intern das Paket `chessfss`. Abbildung 11 zeigt zwei Beispiele. Im ersten werden statt Symbolen (deutsche) Buchstaben für die Figuren verwendet, das zweite zeigt eine meiner lokalen Schriften. In der Dokumentation von `chessfss` ist beschrieben, wie alternative Schriften installiert und verwendet werden.

```
\mainlinestyle \opencommands \beforenumber
1\whiteopen \beforewhite e4!\afterwhite
\beforeblack e5\afterblack \beforenumber
2\whiteopen \beforewhite ♘f3\closecommands
```

```
1 \mainline{1. e4! e5 2. Nf3}
```

```
\mainlinestyle \opencommands \beforenumber
2\blackopen \beforeblack ♘f6\closecommands
```

```
1 \mainline{2... Nf6}
```

```
\variationstyle \opencommands \beforenumber
1\whiteopen \beforewhite e4\afterwhite
\beforeblack e5\afterblack \beforenumber
2\whiteopen \beforewhite ♘f3\closecommands
```

```
1 \variation{1. e4 e5 2. Nf3}
```

```
\variationstyle \opencommands \beforenumber
2\blackopen \beforeblack ♘f6\closecommands
```

```
1 \variation{2... Nf6}
```

```
\variationstyle \opencommands 1. e4 e5 2. ♘f6 ♘f6
\closecommands
```

```
1 \movecomment{1. e4 e5 2.
Nf6 Nf6}
```

```
\variationstyle ♘f3 \variationstyle ... ♘f6
```

```
1 \wmove{Nf3} \bmove{Nf6}
```

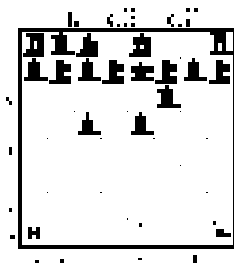
Abbildung 9: Formatierungsbefehle

1 e4 e5 2 ♘f3 ♘c6
3. ♚b5 a6

```
1 \styleB\newgame
2 \mainline{1. e4 e5 2. Nf3 Nc6}
3
4 \makeatletter
5 \renewcommand\styleB@whiteopen {.~}
6 \makeatother
7
8 \styleB
9 \mainline{3. Bb5 a6}
```

Abbildung 10: Änderung des Stils

1. e4 e5 2. Lc4 Lc5 3. Sf3 Sf6



```

1 \setfigtextchars
2   {K}{D}{T}{L}{S}{P}
3 \usetextfig
4 \newgame
5 \mainline{1. e4 e5 2. Bc4 Bc5 3. Nf3 Nf6}

```

```

1 \usesymfig
2 \setfigfontfamily{maya}%
3 \setboardfontfamily{maya}%
4 \mainline{4. Nc3 Qe7}
5 \showboard

```

Abbildung 11: Neue Schriften

Ausblick: offene Baustellen

Man kann mit *skak.sty* sehr schöne Turnierbulletins und Trainingsblätter erstellen. Aber natürlich gibt es Dinge, die verbessert werden könnten:

Mehr Einfluss für den Benutzer

Vieles in *skak.sty* ist mehr oder weniger fest verdrahtet und daher nur veränderbar, wenn man sich in den Code einliest und dann herumtrickst. Der erste Schritt zu mehr Flexibilität – die neue Schriftverwaltung – wurde getan, aber gibt es noch Weiteres zu tun:

- Bei den Brettbefehlen sollte der Benutzer mehr Einfluss auf den Rahmen, auf die Ausmaße der Box und auf die Position der Grundlinie haben.
- Es fehlt ein vernünftiges »Interface« zur Definition und Verwaltung der Stile. Es sollten auch die Möglichkeiten Varianten zu formatieren verbessert werden, z. B. durch eine bessere Zusammenarbeit mit *texmate.sty*.
- Die internen PostScript-Befehle sollten veränderbar und benutzbar sein.

Speicherung von Zügen

Es fehlen weitere Befehle in der Art von `\lastmove`. Nützlich fände ich z.B. `\storemove{<label>}`, `\printmove{<label>}`, `\highlightmove{<label>}`, `\highlightlastmove`.

Die lange algebraische Notation

Es ist derzeit nicht möglich, eine Partie im PGN-Format automatisch in langer Notation auszudrucken. Es gibt aber Hoffnung, dass sich dies demnächst ändert: Bei der Version 1.4 hat sich die Parsingmethode geändert. Während früher zuerst das Argument abgearbeitet wurde, um die Stellung zu aktualisieren, und anschließend das Argument noch mal eingelesen wurde, um es für den Druck zu formatieren, wird jetzt Zug um Zug erkannt und gedruckt. Daher sollte es jetzt prinzipiell möglich sein, beim Drucken die Ergebnisse des Parsens (»woher kam denn die Figur?«) zu verwenden. Der Parsingalgorithmus muss diese Informationen nur vernünftig abspeichern. (Und Torben und ich müssen die Zeit finden, mal darüber nachzudenken.)

Literatur und Software

- [1] <http://sourceforge.net/projects/latex-skak/>
- [2] CTAN:/macros/latex/contrib/chessfss/
- [3] CTAN:/fonts/chess/skaknew/
- [4] <http://de.wikipedia.org/wiki/Forsyth-Edwards-Notation>
- [5] http://de.wikipedia.org/wiki/Portable_Game_Notation
- [6] <http://www.perce.de/LaTeX/pst-pdf/>
- [7] <http://pgn2ltx.sourceforge.net/>
- [8] <http://search.cpan.org/~gmax/Chess-PGN-Parse-0.18/Parse.pm>

Neue Mechanismen in der Verarbeitung von Keys

Hendri Adriaens, Uwe Kern

Dieser Artikel¹ soll in das (L^A)T_EX-Paket `xkeyval` einführen, welches eine Erweiterung des bekannten Pakets `keyval` ist. Das neue Paket bietet flexiblere Kommandos, Syntaxerweiterungen und einen neuen Mechanismus zur Verarbeitung von Klassen- und Paket-Optionen, der sich die Syntax `key=value` zu Nutze macht.

Einführung

Das Paket `keyval` [2] von David Carlisle wird häufig von Paket-Autoren verwendet, um eine große Zahl optionaler Argumente in Makros einzubauen. Die Vorteile von `keyval` sind, dass zum einen die Zahl optionaler Argumente nicht mehr auf neun limitiert ist und dass zum anderen die Argumente Namen erhalten und so die Syntax eines Makros vereinfachen.

Mit dem Paket können so genannte »Key-Makros« definiert werden, die die Benutzereingaben verarbeiten. Durch `\KV@family@keyname` werden diese »Key-Makros« definiert, wobei `KV` ein festes Präfix ist, das Kollisionen mit anderen Paketen vermeidet. Diese Makros haben ein Argument, um Benutzereingaben zu verarbeiten. Ein Makro für den Key `pi` kann z. B. definiert werden über

```
1 \define@key{myfam}{pi}{\setlength{\parindent}{#1}}
```

Diese Zeile definiert ein Makro namens `KV@myfam@pi`. Solche Key-Makros werden aufgerufen, wenn `\setkeys` verwendet wird, um die Keys zu setzen. In unserem Fall wird das Key-Makro `\parindent` auf den entsprechenden Wert gesetzt, wenn `pi` benutzt wird. Hier ein typisches Beispiel für die Anwendung:

```
1 \setkeys{myfam}{pi=10pt,pn=Page~\thepage}
```

¹ Erstveröffentlichung erfolgte in *MAPS* 31 (2004). Dieser um Zusätze erweiterte Nachdruck erfolgt mit freundlicher Genehmigung der Autoren. Die Übersetzung erledigte Johannes Reinhardt.

Die Pakete `keyval` und `xkeyval` richten sich in erster Linie an Klassen- und Paketautoren. Die `\define@key`-Kommandos werden gewöhnlich in der Dokumentenpräambel oder einem Paket verwendet und der Anwender benutzt `\setkeys`. Die Datei `xkeyval.tex` kann mit Plain T_EX verwendet werden. Alle Features, die hier beschrieben werden, sind auch dort verfügbar, mit Ausnahme des »X«-Makros auf S. 49.

Warum ein neues Paket?

Beim Arbeiten an einem anderen Paket entstand der Bedarf, mehrere Familien von Keys zur Verfügung zu haben. Jede Familie sollte Keys für ein spezielles Makro oder eine spezielle Umgebung liefern. So wäre man in der Lage, die Verwendung unzulässiger Keys, die zerstörerische Auswirkungen auf den Rest des Dokuments haben könnten, innerhalb der Makro-Argumente zu verhindern. Nett wäre es auch, dem Benutzer erlauben zu können, gewisse Keys eines Makros oder einer Umgebung global in der Präambel festzulegen. So könnte man beispielsweise die Vorgaben zum Aussehen aller `example`- und `exercise`-Umgebungen des Dokumentes in der Präambel erlauben, aber gleichzeitig verbieten, das Aussehen lokal zu verändern, und umgekehrt. In etwas komplizierteren Fällen kann die Verwendung von Keys in Makros, die für den Umgang mit diesen Keys nicht gedacht sind, zu Fehlern führen, die kaum noch nachzuvollziehen sind. Das war der Anfang des Pakets `xkeyval` [1].

Als wir dabei waren, `keyval` zu verallgemeinern, bemerkten wir, dass es bereits eine Menge Pakete gab, die gewisse Fähigkeiten erweitern, alle auf ihre eigene Art. Einige liefern z. B. ein System für den Gebrauch von Keys und Werten in `\usepackage`-Kommandos. Die bekanntesten Beispiele sind die Pakete `hyperref`, `geometry` und `beamer`. All diese Ansätze unterscheiden sich in Details und lassen sich nicht ohne weiteres Programmieren auf andere Pakete übertragen. Das verlangt nach einer einheitlichen Lösung.

Eine besondere Fähigkeit, die beispielsweise in `hyperref` verwendet wurde, ist die Verfügbarkeit von Keys des Typs `boolean`, die nur die Werte `true` oder `false` annehmen können. `hyperref` implementiert das über `\define@key` im gewöhnlichen Key-System. Weil aber die Funktion (oder zumindest ein Teil), die bei Verwendung des Keys aufgerufen werden soll, schon vorher bekannt ist (nämlich das einfache Setzen eines »if«-Kommandos je nach Aufruf auf `true` oder `false`), kann das System vereinfacht werden.

Ein letzter Grund, das neue Paket zu entwickeln, war die Tatsache, dass die Entwicklung des Pakets `keyval` offenbar seit 1999 stillstand und fundamentale Änderungen und Verbesserungen am System mit einem neuen Paket einfacher zu realisieren waren. Die Verbesserungen betreffen unter anderem Makros, die Paket-Optionen erzeugen, mit denen Werte übergeben werden können. Es gibt neue Arten von Keys, eine Zeiger-Syntax und ein System für vordefinierte Werte. Mehrere Familien in `\setkeys` und robustes Eingabe-Parsing werden akzeptiert. Schließlich werden die Pakete aus der `PSTricks`-Serie unterstützt. In den folgenden Abschnitten dieses Artikels werden diese Entwicklungen behandelt.

Keys und Werte in Paket-Optionen

Zu allererst stellt das Paket Makros zur Verfügung, um Klassen- oder Paket-Optionen zu deklarieren, auszuführen und zu verarbeiten. Die Makros sind unter den üblichen $\text{\LaTeX}$ -Namen verfügbar, haben aber alle das Suffix `X`, nämlich

```
1 \DeclareOptionX
2 \DeclareOptionX*
3 \ExecuteOptionsX
4 \ProcessOptionsX
```

Mit diesen Kommandos kann der Benutzer einer Option einen Wert zuordnen, genauso wie mit `\setkeys`. Das erste Makro basiert auf `\define@key`, die letzten beiden basieren auf `\setkeys`. Wenn das Paket `mypack` die Befehle kennt, könnte ein Benutzer beispielsweise Folgendes machen:

```
1 \usepackage[textcolor=red,font=times]{mypack}
```

Diese Makros sind vollständig in das System der $\text{\LaTeX}$ -Optionen integriert. So können Pakete globale Optionen aus dem `\documentclass`-Befehl kopieren, Optionen an andere Klassen oder Pakete weiterreichen und die Liste der unbenutzten, globalen Optionen aktualisieren, die dann von $\text{\LaTeX}$ im Logfile ausgegeben wird.

Wertzuweisungen wie `author=\textit{Me}` in Klassen- und Paket-Optionen sind jedoch nicht erlaubt, obwohl sie leicht von `\setkeys` verarbeitet werden könnten. Diese Einschränkung resultiert aus der Art, wie $\text{\LaTeX}$ Optionen verarbeitet. Dabei wird die gesamte Liste von Optionen (Keys und

Werte) komplett expandiert, was offensichtlich Probleme verursacht, denn `author=\protect\textit{Me}` wäre *keine* Lösung für dieses Problem.

Um diese verfrühte Expansion zu vermeiden, müssen diverse Kernel-Makros undefiniert werden. `xkeyval` beinhaltet das Paket `xkvltxp`, welches die neuen Definitionen enthält. Lädt man das Paket bevor man die Klassen oder Pakete einbindet, die `xkeyval` nutzen, so können die Klassen- und Paket-Optionen expandierbare Makros enthalten. `xkvltxp` wird nicht in den L^AT_EX 2_ε-Kernel eingebaut, weil es Kompatibilitätsschwierigkeiten bei Anwendern verursachen könnte, die einen alten Kernel zusammen mit neuen Paketen benutzen, die diese veränderte Funktionalität voraussetzen.

Präfixe, Familien, Keys und Zeiger

Das Paket `xkeyval` liefert eine erweiterte Syntax für alle Befehle aus `keyval`. Die Dokumentation zu `xkeyval` erläutert syntaktische Details, die in diesem Artikel nicht diskutiert werden. Der Befehl, mit dem Keys definiert werden, wurde um ein optionales Argument erweitert, das dem Key-Makro als Präfix vorangestellt wird. Alle Paket-Autoren sollten ein paketspezifisches Präfix für alle internen Makros verwenden, um nicht versehentlich Makros anderer Pakete umzudefinieren. Mit diesem Argument können auch in spezialisierten Systemen wie `PSTricks` Keys definiert und gesetzt werden. Darauf werden wir im Abschnitt über das Paket `pst-xkey` zurückkommen.

Die Syntax für das Setzen von Keys mit `\setkeys` wurde entsprechend angepasst. Wie in der Einführung erwähnt, kann man auch eine Liste von Familien angeben, die beim Setzen von Keys durchsucht werden sollen:

```
1 \setkeys{font,page}{fs=10pt,pn=Page~\thepage}
```

Das Paket bietet auch neue Typen von Keys. Dabei handelt es sich um Auswahl-Keys, die eine beschränkte Zahl möglicher Eingabewerte zulassen und um boolesche Keys, die ausschließlich die Werte `true` und `false` annehmen können, wie im folgenden Beispiel:

```
1 \define@choicekey{fam}{keya}{\fbox,\mbox}{#1{text}}
2 \define@boolkey{fam}{keyb}{%
3   \ifKV@fam@keya we continue\else we stop here\fi}
4 \setkeys{fam}{keya=\mbox,keyb=false}
```

Diese Keys erzeugen eine Fehlermeldung, wenn der Benutzer einen Wert bestimmt, der nicht erlaubt ist. Weitere Erläuterungen dazu enthält die Dokumentation. Das Paket bietet mit `xkvview` ein Werkzeug, um Tabellen mit Informationen über die definierten Keys zu erstellen.

Mit der neuen Syntax können auch Zeiger auf Keys verweisen. Mit Zeigern kann man z. B. `keyb` den Wert zuweisen, der `keya` zugewiesen wurde, völlig egal, was dieser Wert ist. Beispielsweise

```
1 \setkeys{family}{\savevalue{keya}=red,keyb=\usevalue{keya}}
```

`\savevalue` veranlasst an dieser Stelle `xkeyval` dazu, sich den Wert zu merken, der `keya` zugewiesen wurde. `\usevalue` verwendet diesen Wert. (Man kann sich mit dem `\savekeys`-Befehl das wiederholte Tippen von `\savevalue` sparen.) Wenn in diesem Beispiel `red` auf `blue` umgestellt wird, so müssen keine Änderungen mehr an `keyb` vorgenommen werden, damit diesem ebenfalls `blue` zugewiesen wird. Ähnlich verarbeitet $\text{\TeX}$ die Definition `\def\cmdb{\cmda}`.

Diese Zeiger können auch für das Setzen von Standardwerten verwendet werden. Sie weisen dem Key-Makro einen Standardwert zu, sobald der Benutzer einen bestimmten Key gebraucht, ihm aber keinen Wert zuweist. So könnte man z. B. definieren:

```
1 \define@key{fam}{keya}{keya: #1 }
2 \define@key{fam}{keyb}{\usevalue{keya}}{keyb: #1 }
```

Dann würde der folgende Aufruf von `\setkeys`

```
1 \setkeys{fam}{\savevalue{keya}=test,keyb}
```

das unten stehende Ergebnis liefern:

```
1 keya: test keyb: test
```

Wir werden nun ein paar technische Details zur Syntax der Zeiger diskutieren. Zu Beginn sind die Kontrollsequenzen `\savevalue` und `\usevalue` noch nicht definiert! Statt dessen nutzt das Paket sie als Platzhalter. Ein einfacher Parsing-Schritt entscheidet, ob `\savevalue` im Namensteil des Keys benutzt wurde. Parsing wird auch genutzt, um das Auftreten von `\usevalue` durch den gespeicherten Wert zu ersetzen. Wird ein Zeiger ersetzt, so wird dieser

Ersatz wiederum auf Zeiger durchsucht, die gegebenenfalls ersetzt werden. Das ermöglicht verschachtelte Zeiger in Key-Werten. Außerdem stellt dieses Parsing sicher, dass ein Wert keine Zeiger mehr enthält, wenn er dem Key-Makro übermittelt wird – solange der Zeiger nicht in einer Gruppe vor `xkeyval` versteckt wurde.

Wenn der Benutzer dem Key keinen Wert zugewiesen hat, ist der Ersetzungsprozess ein bisschen vertrackter. In diesem Fall soll der Standardwert des Keys (wenn vorhanden) nach Zeigern durchsucht werden. Standardwert-Makros sehen so aus:

```
1 \def\prefix@fam@key@default{\prefix@fam@key{the default value}}
```

Das Makro `\prefix@fam@key@default` wird ausgeführt, wenn der Benutzer dem Key keinen Wert zuweist. Dieses System wurde von `keyval` eingeführt und viele Pakete machen davon Gebrauch. Einige Pakete nutzen es aber nicht auf die Weise, wie es in `keyval` gedacht ist. Zum Beispiel definiert das Paket `fancyvrb` Standardwert-Makros, die beliebigen Code an Stelle des vorgesehenen `\prefix@fam@key` ausführen. Um die Kompatibilität zu gewährleisten, muss dieses Verhalten unterstützt werden, obwohl dadurch eine saubere Lösung des Problems verbaut ist. Beispielsweise könnten wir sonst `\prefix@fam@key@default` von Beginn an als »den Standardwert« definieren – ohne getrennten Makroaufruf.

Das ist eine wichtige Einschränkung des Zeiger-Systems, da wir den Standardwert vom Standardwert-Makro erhalten und es nach Zeigern absuchen wollen. Deshalb verfährt `xkeyval` folgendermaßen: Es prüft zuerst, ob das Standardwert-Makro wie erwartet startet, nämlich mit einem Key-Makro `\prefix@fam@key`. Ist das der Fall, definiert es lokal das Key-Makro um und sichert den Wert in einem temporären Makro, um es anschließend auszuführen. Das temporäre Makro enthält dann den Standardwert, der nach Zeigern durchsucht werden kann. Ist das Standardwert-Makro nicht von der erwarteten Art, wie bei `fancyvrb`, dann führt `xkeyval` es aus, ohne zu versuchen, den Standardwert zu erhalten oder Zeiger zu ersetzen.

Voreingestellte Werte

Das System der Standardwerte kommt zum Einsatz, wenn Benutzer Keys angeben, diesen Keys aber keine Werte mitgeben. Das Paket `keyval` bietet ursprünglich keinen Weg, um auch solchen Keys, die überhaupt nicht vom

Benutzer verwendet werden, einen Wert zuzuweisen. In vielen Anwendungsfällen will man Standardwerte für Keys einsetzen, wenn diese nicht explizit benutzt werden. Beispielsweise »Skaliere diese Abbildung mit Faktor 1 wenn der Benutzer nichts anderes sagt«. Nun könnte man einfach loslegen, das Key-Makro mit einer Voreinstellung aufrufen und nachher die Benutzereingabe an `\setkeys` weiterleiten, um schließlich die Werte, die man gerade gesetzt hat, wieder zu überschreiben. In Fällen, wo Keys nicht selbstständig Informationen generieren, sondern z. B. nur eine Länge setzen, ist das möglich (aber recht schwerfällig, wenn man viele Keys hat).

Was passiert aber, wenn wir dieses Schema auf Keys anwenden, die folgendermaßen definiert sind?

```
1 \define@key{fam}{keya}{Your input was: #1}
2 \define@key{fam}{keyb}{\edef\list{\list,#1}}
```

Wenn wir dem selben Schema folgen wie im ersten Beispiel, so würde sowohl unsere Voreinstellung als auch unsere Benutzereingabe (wenn vorhanden) in der Ausgabe erscheinen. Im zweiten Beispiel werden die Voreinstellung und die Benutzereingabe an die Liste in `\list` angehängt.

Um das zu vermeiden, verwaltet `xkeyval` Voreinstellungen. Zuerst deklariert man die Keys, die voreingestellt werden sollen, und ihre Werte über `\presetkeys`, z. B.:

```
1 \savekeys{fam}{head}
2 \presetkeys{fam}{head=red}{tail=\usevalue{head}}
```

Der Grund, weshalb das Makro `\presetkeys` zwei Argumente hat, die Voreinstellungen für Keys enthalten, wird gleich verständlich werden.

Wenn jetzt Benutzereingaben die Keys in der Familie `fam` festlegen, stellt das Makro `\setkeys` fest, welche Keys vom Benutzer gesetzt werden und füllt diese *nicht* mit Voreinstellungen. Keys, die der Benutzer nicht festlegt, werden dann auf die Werte gesetzt, die in `\presetkeys` eingestellt sind.

Wenn man Zeiger einsetzt, sollte man eine Sache unbedingt im Hinterkopf behalten: Zeigt der Zeiger auf einen Key, der erst später einen Wert zugewiesen bekommt, so kann der Zeiger den Wert zu diesem Zeitpunkt nicht wissen und es kommt zu Fehlern. Daher ist es meistens, wie in dem Beispiel oben, am besten, Voreinstellungen mit Zeigern erst ganz am Ende auszuführen.

Eine ähnliche Unstimmigkeit kann auftreten, wenn Keys ohne Zeiger in ihrem Wert *nach* dem Setzen der Benutzereingaben eine Voreinstellung zugewiesen bekommen. Benutzer können dann im späteren Verlauf der Ausführung keine Zeiger mehr auf diese Voreinstellungen verwenden. Deshalb sollte man Keys ohne Zeiger mit ihrem Wert ganz am Anfang ausführen, noch bevor Benutzereingaben gesetzt werden.

Das ist auch der Grund, warum das Makro `\presetkeys` zwei Argumente hat: Das erste (das gewöhnlich Keys und Werte ohne Zeiger enthält) wird eingefügt, bevor vom Benutzer eingegebene Keys gesetzt werden, das zweite (das Zeiger zu Voreinstellungen oder Benutzereingaben enthält) danach.

Dieses Vorgehen ist insbesondere dann nützlich, wenn man sich nicht darauf verlassen kann, dass Werte von Keys lokal in einem Makro oder einer Umgebung bleiben, weil das System für Voreinstellungen die Werte der Keys bei jedem Aufruf des Makros oder der Umgebung auf die Voreinstellung zurücksetzt, wenn der Benutzer diese nicht lokal überschrieben hat. Wir werden das näher erläutern: `\def`-Definitionen (beispielsweise erstellt von Key-Makros) werden von `TEX` beim Verlassen einer Gruppe oder Umgebung zerstört. Deshalb bleiben solche Werte lokal. Wenn allerdings die Keys nicht immer `\def` benutzen, sondern z. B. `\gdef`, dann durchbrechen diese globalen Definitionen ihre Gruppe oder Umgebung und können alle folgenden Makros und Umgebungen durcheinander bringen. Daher muss man die Werte der Keys bei jedem Gebrauch des Makros oder der Umgebung sorgfältig neu initialisieren.

All das ist mit dem System der Voreinstellungen nicht mehr notwendig. Wenn erst einmal die Voreinstellungen für eine bestimmte Familie definiert wurden, werden jedes Mal, wenn die Familie in `\setkeys` verwendet wird, die Voreinstellungen zusammen mit den Benutzereingaben berücksichtigt.

Das folgende Beispiel zeigt die Stärke einer Kombination von Voreinstellungen und Zeigern. Unter dem Beispiel folgt die Ausgabe und eine Erklärung. Nehmen wir an, dass wir ein einfaches Kommando für eine Box mit Rahmen und Schatten mit folgendem *Standard*verhalten erstellen wollen:

- Ein Schatten wird nur dann gezeichnet, wenn die Box einen Rahmen hat.
- Die Schattenfarbe soll 40 % der Farbe des Rahmens haben, um klar unterscheidbar zu sein.
- Der Schatten soll viermal so breit wie der Rahmen sein.

Selbstverständlich soll der Benutzer jede dieser Regeln überschreiben können, wenn er das Box-Kommando verwendet.

```

1 \documentclass{article}
2 \usepackage{xkeyval}
3 \usepackage{calc,xcolor}
4 \makeatletter
5 \newdimen\shadowsize
6 \define@boolkey{Fbox}{frame}[true]{}
7 \define@boolkey{Fbox}{shadow}[true]{}
8 \define@key{Fbox}{framecolor}{\def\Fboxframecolor{#1}}
9 \define@key{Fbox}{shadowcolor}{\def\Fboxshadowcolor{#1}}
10 \define@key{Fbox}{framesize}{\setlength\Fboxrule{#1}}
11 \define@key{Fbox}{shadowsize}{\setlength\shadowsize{#1}}
12 \savekeys{Fbox}{frame,framecolor,framesize}
13 \presetkeys{Fbox}{frame,framecolor=black,framesize=0.5pt}%
14 {shadow=\usevalue{frame},shadowcolor=\usevalue{framecolor}!40,
15 shadowsize=\usevalue{framesize}*4}
16 \newcommand*\Fbox[2][{}]{%
17 \setkeys{Fbox}{#1}{\ifKV@Fbox@frame\else\Fboxrule0pt\fi
18 \ifKV@Fbox@shadow\else\shadowsize0pt\fi
19 \sbox0{\fcolorbox{\Fboxframecolor}{white}{#2}}%
20 \hskip\shadowsize
21 \color{\Fboxshadowcolor}\rule[-\dp0]{\wd0}{\ht0+\dp0}%
22 \llap{\raisebox{\shadowsize}{\box0\hskip\shadowsize}}}%
23 \makeatother
24 \begin{document}
25 \Fbox{demo1} \Fbox[framecolor=gray]{demo2}
26 \Fbox[shadow=false]{demo3} \Fbox[framesize=1pt]{demo4}
27 \Fbox[frame=false,shadow]{demo5}
28 \end{document}

```

demo1 demo2 demo3 **demo4** demo5

Zuerst werden in den Zeilen 6 bis 11 die Keys definiert, die wir in diesem Beispiel brauchen. Das Kommando `\presetkeys` in Zeile 13 definiert die Voreinstellungen: Der Rahmen wird auf `true` gesetzt, seine Farbe auf Schwarz und seine Rahmengröße auf 0.5 pt, es sei denn, der Benutzer liefert andere Einstellungen zu diesen Keys. Die oben aufgelisteten Anforderungen werden durch die Zeiger-Ausdrücke im nächsten Argument abgedeckt.

Die erste Anwendung der Box zeigt die Standard-Box ohne zusätzliche Benutzereingaben. Wir sehen einen Rahmen und einen Schatten, alles basierend auf der Farbe schwarz. Die zweite Box zeigt, wie die Benutzereingaben für den Rahmen die voreingestellten Werte überschreiben. Die Box wird grau. Weil aber die Schattenfarbe standardmäßig von der Rahmenfarbe abgeleitet wird, wird der Schatten hellgrau. Im dritten Beispiel haben wir einen Rahmen, aber keinen Schatten. Zu beachten ist, dass die Rahmenfarbe wieder schwarz – die Voreinstellung – ist. Die vierte Box hat eine erhöhte Rahmenbreite und durch den Gebrauch des Zeigers beim Voreinstellen der Keys auch eine größere Schattenbreite. Das letzte Beispiel zeigt, dass es möglich ist, das voreingestellte Verhalten zur Verbindung zwischen Rahmen und Schatten vollkommen zu überschreiben: Es gibt einen Schatten ohne Rahmen aus.

Robustes Parsen

Genau wie die Zeiger-Platzhalter `\savevalue` und `\usevalue`, behandeln `keyval` und `xkeyval` auch das Komma und das Gleichheitszeichen als Platzhalter. In der Vergangenheit hat das zu Problemen geführt. So gibt es eine bekannte Inkompatibilität zwischen den Einstellungen des Pakets `babel` für die türkische Sprache und allen Paketen, die `keyval` benutzen. Weil das türkische `babel` den Catcode des Gleichheitszeichens zu Gunsten einer Abkürzung verändert, können die Parser-Makros aus `keyval` diese Zeichen nicht mehr erkennen und produzieren Fehlermeldungen. Weitere Information zu diesem Problem mit `keyval` und `babel` gibt es bei <http://www.latex-project.org/cgi-bin/ltxbugs2html?pr=babel/3523>.

`xkeyval` löst dieses Problem, indem es alle beim Parsen benötigten Zeichen säubert (d. h. den catcode auf 12 setzt). Diese Aufgabe übernimmt das Makro `\@selective@sanitize`, das ein einzelnes oder mehrere Zeichen in einem einzigen Aufruf säubern kann. Auch die Tiefe der Säuberung kann angegeben werden. `xkeyval` implementiert das Makro so, dass nur Kommas und Gleichheitszeichen in der obersten Ebene eines Key-Werts gesäubert werden. Das ist schließlich alles, was man braucht, um die Eingabe zu parsen. Zeichen innerhalb von Gruppen bleiben unberührt, und man kann deshalb sogar `babel`-Kurznotationen verwenden ohne Fehler zu produzieren:

```
1 \usepackage[turkish]{babel}
2 ... \setkeys{fam}{key={some =text}}
```

In diesem Beispiel wird das erste »=« zum Parsen gesäubert, während das zweite »=« unberührt bleibt und damit seine ursprüngliche Bedeutung behält.

Makros umdefinieren?

Bestehende Makros umzudefinieren ist im Allgemeinen gefährlich. Trotzdem definiert das Paket `xkeyval` die zwei `keyval`-Hauptmakros `\define@key` und `\setkeys` um. So verhindern wir, dass verschiedene, parallel laufende Systeme ähnliche Dinge tun und Verwirrung stiften.

Obwohl `xkeyval` die gesamte Syntax des Originalpakets `keyval` unterstützt, mussten wir die Pakete prüfen, die `keyval` benutzen, bevor wir uns entscheiden konnten, die Makros umzudefinieren. Drei Hauptfragen kamen in diesem Prozess auf.

Zuerst mussten wir feststellen, dass ein paar Pakete interne Teile von `keyval` statt der Benutzerschnittstelle über `\define@key` und `\setkeys` verwenden. Um jegliche Fehler wegen nicht definierter Kontrollsequenzen zu vermeiden, lädt `xkeyval` also die Interna von `keyval`, wenn `keyval` nicht schon vorher geladen wurde. Zweitens benutzten bestimmte Pakete das Standardwert-System auf kreative Weise. Im Abschnitt über die Zeiger-Syntax haben wir das Problem dargestellt und `xkeyval`s Lösung erläutert.

Schließlich haben wir entdeckt, dass `pst-key` die Makros `\define@key` und `\setkeys` umdefinierte, um das Setzen von Keys in `PSTricks` zu ermöglichen. Wir haben diesen Punkt mit Herbert Voß, dem Maintainer von `PSTricks` diskutiert, und sind zu dem Schluss gekommen, dass `xkeyval` einen einheitlichen Zugriff auf Keys und Werte entwickeln sollte, um so `pst-key` abzulösen. Mehr Informationen über die Entwicklung von `PSTricks` gibt es im letzten Abschnitt.

Nachdem die nötigen Makros umdefiniert sind, stellt `xkeyval` sicher, dass das Paket `keyval` nicht nachträglich geladen werden kann. So wird verhindert, dass die `xkeyval` Makros noch einmal umdefiniert werden. Dies ist der letzte Schritt, um die `keyval`-Makros sicher umzudefinieren und ein System anzubieten, auf das alle Paket-Autoren ihre Pakete ohne allzu großen Aufwand umstellen können.

Das Paket `pst-xkey`

Eine entscheidende Reihe von Paketen wird in naher Zukunft `xkeyval` benutzen, nämlich die `PSTricks`-Pakete [3, 4]. Für das Verarbeiten von Keys und

Werten verwenden sie momentan noch eine Kombination aus eigenen Definitionen in `pstricks.tex` und `pst-key`, wobei letzteres eine Modifikation des Pakets `keyval` darstellt.

Wegen der Beliebtheit und Flexibilität des Pakets `PSTricks` haben mehrere Nutzer Erweiterungen zur Originaldistribution beigetragen. Unglücklicherweise hatten alle Keys in `PSTricks` dieselbe Form, nämlich `\psset@somekey`. Deshalb mussten `PSTricks`-Autoren alle bestehenden Pakete durchsehen, um nicht irgendeinen bestehenden Key umzudefinieren.

Herbert Voß hatte das Problem erkannt und bald begann die Arbeit, `xkeyval` den Weg zu ebnen, um damit in `PSTricks`-Keys zu definieren und zu setzen. Wenn Paket-Autoren ihre Keys in eine ausgewählte Familie (z.B. mit dem Paketnamen) verpacken, dann brauchen sie keine anderen Pakete nach bestehenden Keys abzusuchen. Darin liegt der entscheidende Vorteil von `xkeyval`.

Um das zu ermöglichen, mussten `\define@key` und `\setkeys` so angepasst werden, dass der Standardpräfix `KV` von `keyval` verändert werden kann, z. B. in `psset`. Dann musste das Makro `psset` undefiniert werden, um das neue `\setkeys` zu benutzen und alle verfügbaren Familien zu suchen. Wenn ein Paket aus `PSTricks` geladen wird, fügt es alle Familien, die in diesem Paket benutzt werden, einer Liste hinzu. Diese Liste wird dann in `\setkeys` verwendet. Weil alle unterschiedlichen Pakete verschiedene Familien benutzen, ist die Wiederverwendung von Namen für Keys bald kein Problem mehr. Die neue Definition von `\psset`, ebenso wie ein paar andere Makros, die für diese Aufgabe benötigt werden, ist im Paket `pst-xkey` enthalten, das zusammen mit `xkeyval` verteilt wird.

Literatur

- [1] Hendri Adriaens: *xkeyval package, v2.4, 2005/03/31*; CTAN:/macros/latex/contrib/xkeyval.
- [2] David Carlisle: *keyval package, v1.13, 1999/03/16*; CTAN:/macros/latex/required/graphics.
- [3] Herbert Voß: *PSTricks web site*; <http://www.pstricks.de>.
- [4] Timothy Van Zandt et al.: *PSTricks package, v1.04, 2004/06/22*; CTAN:/graphics/pstricks.

writer2latex: Eine Praxiserfahrung

Michael Niedermair

Dieser Artikel beschreibt, wie ein Buch, das mit MS-Word geschrieben worden ist, mit Hilfe von *writer2latex* in L^AT_EX umgesetzt wurde, welche Probleme dabei auftraten und wie sie gelöst wurden.

Einleitung

writer2latex ist ein Java-Programm, das OpenOffice.org-Dokumente (OOo) nach L^AT_EX umsetzt. OOo ist in der Lage, MS-Word-Dokumente zu lesen und diese im OOo-Format abzuspeichern, so dass *writer2latex* angewendet werden kann. Unter der Adresse <http://www.hj-gym.dk/~hj/writer2latex/> kann *writer2latex* bezogen werden. Zusätzlich ist eine installierte Java-Umgebung notwendig.

MS-Word-Dateien umwandeln

Für jedes Kapitel des Buches wurde eine eigene MS-Word-Datei verwendet (insgesamt 19 Dateien). Alle Bilder waren dabei als externe Bilder mit einem Verweis eingebunden und lagen als TIFF-Datei vor. Im Änderungsmodus wurden dabei Veränderungen vom Fach- und Sprachlektorat durchgeführt. In diesem Änderungsmodus werden alle Änderungen farblich hervorgehoben (jeder Bearbeiter bekommt eine andere Farbe) und mit dem Kürzel des Bearbeiters versehen. Somit lässt sich schnell erkennen, wo was von wem geändert worden ist. Änderungen können dann beliebig akzeptiert oder verworfen werden. Alle diese Änderungen wurden global durchgeführt und die Datei anschließend gespeichert. In OOo wurde dann die MS-Word-Datei geöffnet und im OOo-Format (SXW-Datei) gespeichert.

Später fiel auf, dass einige Änderungen (über den Änderungsmodus) im Index nicht korrekt durchgeführt worden waren, so dass alter und neuer Wert in L^AT_EX übernommen worden sind (zum Beispiel `\index{FurbeFarbe}`). Dieses Problem ließ sich lösen, indem in MS-Word nicht normal, sondern mit »Speichern unter« unter einem neuen Dateinamen abgespeichert wurde. Bei »normalen« Speichern werden die Änderungen meist am Ende der Datei angehängt, gelöschte Bereiche werden nur markiert usw. Dies hat den Effekt,

dass die Datei immer größer werden kann, bis MS-Word sich entschließt, die nicht mehr notwendigen Bereiche komplett zu entfernen. Mit »Speichern unter« werden wirklich nur die Bereiche gespeichert, die noch aktuell sind. Aller unnötiger Ballast verschwindet dabei. Ob das Problem bei MS-Word oder OOo lag, ließ sich nicht genau feststellen.

Konfiguration von *writer2latex*

writer2latex wird über eine Konfigurationsdatei (eine XML-Datei) konfiguriert. Da alle Formatierungen bis auf die Layoutvorlagen ignoriert werden sollen, ist die Konfigurationsdatei *ultraclean.xml* eine gute Vorlage, wobei es öfters vorkommt, dass in einem Absatz die Schrift oder die Schriftgröße vom Autor (meist versehentlich) geändert oder ein anderes Aufzählungssymbol für Listen verwendet wird. Diese Layoutvorlage wird nach *buch.xml* kopiert und entsprechend angepasst. In der Paketdokumentation zu *writer2latex* findet sich eine englische Dokumentation und in [1] eine ausführliche deutsche Beschreibung.

In der Dokumentenvorlage wurden zusätzliche Layouts (*Listing Klein*, *Bild x.x*, ...) definiert. Für diese wurden entsprechende Erweiterungen in der Konfiguration eingefügt:

```

1  <custom-preamble>
2    \usepackage[expert]{lucidabr}
3    \usepackage{listings}
4    \lstloadlanguages{java,HTML}
5    \definecolor{source}{gray}{0.92}
6    \lstset{extendedchars=true,
7            basicstyle=\ttfamily\footnotesize,
8            backgroundcolor=\color{source},
9            captionpos=b,
10           usekeywordsintag=true,
11           breaklines=true,
12           numbersep=2pt,
13           numberstyle=\tiny}
14 </custom-preamble>
15 <!--
16   Legt fest, welche zusätzlichen Pakete, Befehle usw.
17   in das TeX-Dokument geschrieben werden.
18   -->
19 <style-map name="Listing Klein" class="paragraph-block"
20   next="Listing Klein" before="\begin{lstlisting}"

```

```

21   after="\end{lstlisting}" />
22   <style-map name="Listing Klein" class="paragraph"
23     before="" after="" verbatim="true" />
24   <!--
25     Es wird der Inhalt der Formatvorlage 'Listing'
26     (kompletter Bereich) in eine lstlisting-Umgebung
27     gepackt. Dabei wird nur der reine Inhalt
28     (ohne Formatierungen usw.) übernommen.
29     Achtung: Dazu muss das Paket 'listings.sty'
30     bei 'custom-preamble' angegeben werden.
31   -->
32   <style-map name="Bild x.x" class="paragraph"
33     before="\begin{figure}\caption{"
34     after="}\end{figure}" />
35   <style-map name="Gliederung 4" class="paragraph"
36     before="\minisec{" after="}" />

```

Ein Problem ergab sich dadurch, dass die Bildunterschrift nicht als normale Bildunterschrift beim Bild definiert worden ist, sondern mit einem eigenen Layout, welches nachfolgendes Ergebnis erbrachte:

```

1  {\sffamily\bfseries
2   [Warning: Image ignored] % Unhandled or unsupported graphics:
3   %\includegraphics[width=12.487cm,height=5.569cm]{kapitel01_003.tif
4   }
5   }
6   %
7   \begin{figure}\caption{
8   Dahinter steckt das Script \textit{start.js}
9   }\end{figure}

```

Da pdfL^AT_EX das TIFF-Format nicht unterstützt, mussten zuerst alle Bilder mit dem Programm *convert* in das PNG-Format umgewandelt werden. Zusätzlich wurden von Hand die entsprechenden Zeilen (siehe obigen Code) nachgearbeitet. Dabei wurde der Befehl `\includegraphics` in die *figure*-Umgebung verschoben, die Angabe zur Breite und die Dateiendung *.tif* entfernt, so dass das Ergebnis wie folgt aussah:

```

1  \begin{figure}
2   \includegraphics{kapitel01_003}
3   \caption{Dahinter steckt das Script \textit{start.js}}
4  \end{figure}

```

Dieses und andere Probleme wurden dem Autor Henrik Just mitgeteilt und er bemüht sich, eine Lösung bei der nächsten Version einzubauen. An dieser Stelle ein großes Dankeschön an Henrik für die tolle Unterstützung.

Der Aufruf von *writer2latex* sieht dann wie folgt aus:

```
1 w2l -config buch.xml kapitel01.sxw
```

Als Ergebnis wird eine Datei mit dem Namen *kapitel01.tex* erzeugt. Diese Datei hat einen entsprechenden L^AT_EX-Header und kann direkt mit pdfL^AT_EX übersetzt werden. Da dieser Header für die 19 Dateien nicht sinnvoll ist, bietet *writer2latex* eine Option, um den kompletten Header auszublenden:

```
1 <option name="no_preamble" value="true" />
2 <!--
3   Legt fest, ob das TeX-Dokument eine Präambel haben
4   soll. Bei 'true' wird alles bis einschließlich
5   '\begin{document}' und am Ende '\end{document}'
6   weggelassen, so dass das Dokument als Teildokument
7   mit 'input' bzw. 'include' eingebunden werden
8   kann.
9   -->
```

Es wurde also der Header einmal nach *buch.tex* kopiert, und dann wurden alle weiteren Kapitel mit `\include` eingebunden.

Besonderheiten

Hat man den erzeugten T_EX-Code angeschaut, so sind einige »Besonderheiten« aufgefallen. Bei Auflistungen (`itemize`) wird vorher eine entsprechende Anweisung gesetzt, die u. a. festlegt, welches Symbol verwendet werden soll. Diese Anweisung wird im Header definiert und kann entsprechend umgestellt werden.

```
1 \liststyleWviiiNumiii
2 \begin{itemize}
3 \item ...
```

Die Anweisungen wurden durch ein globales »Search and Replace« entfernt, um zu verhindern, dass der Autor eigene Symbole einsetzen kann. Auch hier

soll ein Konfigurationsparameter entstehen, der festlegt, dass diese Anweisung nicht gesetzt wird.

Ähnlich verhält es sich bei `enumerate`-Auflistungen:

```

1 \liststyleWviiiNumx
2 \setcounter{saveenum}{\value{enumi}}
3 \begin{enumerate}
4 \setcounter{enumi}{\value{saveenum}}
5 \item ...

```

Hier wird auch eine entsprechende Anweisung für das Format gesetzt und zusätzlich, wenn die Auflistung fortgeführt worden ist, die Aufzählungsnummer gesetzt. Leider hat hier MS-Word mit der richtigen Vergabe von Nummern etwas Probleme, da diese falsch durchgezählt wurden bzw. auch Nummern ausgelassen worden sind. Auch hier wurde der zugehörige $\TeX$ -Code entfernt.

Eine weitere Besonderheit waren Tabellen. Sie wurden ausschließlich als `longtable`-Tabellen umgesetzt. Mit Hand wurden daraus `table`- und `tabular`-Umgebungen gemacht. Auch dafür ist eine entsprechende Konfiguration geplant, damit die Umgebung ausgewählt werden kann.

Beim kursiven Text und bei der Verwendung des Index ist aufgefallen, dass leere Anweisungen (z. B. `\textit{}`) erzeugt werden.

```

1 Direkt zum aktuellen Dokument gehören die Einstellungen aus dem
   Dialogfenster \textit{Datei}-Informationen}%
2 \index{Datei}-Informationen!DocumentInfo}\textit{}
3 (\textstyleListingZeichen{DocumentInfo}).

```

Weitere Wünsche

Bei der Umsetzung sind noch weitere Wünsche aufgekommen. Es hat sich als sinnvoll erwiesen, eine weitere Funktion einzubauen, die auf bestimmte Texte reagiert und diese umsetzt. So wäre es möglich, aus »z.B.« ein »z.\,B.« zu machen.

Bei einigen Listings fiel auf, dass sie unterschiedlich eingerückt werden, wenn sie vorher mit Tabulatoren oder mit Leerzeichen eingerückt waren. Auch hier wäre eine automatische Umsetzung von Tabulatoren in eine bestimmte Anzahl Leerzeichen schön. Eine Umgehung des Problems lässt sich mit der

`lstlisting`-Umgebung realisieren, indem dort für die Tabulatoren eine bestimmte Anzahl von Leerzeichen definiert wird. Geschickter ist es natürlich, wenn gleich vom Autor nur Leerzeichen verwendet werden.

Ein weiterer Wunsch an mich selbst ist ein OOo-Makro, welches automatisch MS-Word-Dateien einliest, die Änderungen im Änderungsmodus automatisch akzeptiert und die Datei im SXW-Format wieder abspeichert. Und dies bitte noch als Aufruf über die Kommandozeile, so dass diese Funktion in einer *make*-Datei verwendet werden kann.

Ausblick

Bei der neu geplanten OOo-Version 2 soll *writer2latex* ein fest integrierter Bestandteil werden, so dass die Umwandlung direkt von OOo gestartet werden kann.

Zusammenfassung

Das Projekt hat gezeigt, dass es mit nur geringem Aufwand möglich ist, MS-Word-Dokumente nach $\text{\LaTeX}$ umzusetzen und somit wesentlich schöner zu setzen. Mit den Änderungen und Ergänzungen in der nächsten *writer2latex*-Version müsste es sogar möglich sein, unter Verwendung einer entsprechenden Formatvorlage eine vollautomatische Umsetzung durchzuführen, bei der keine Anpassungen per Hand mehr notwendig sind.

Es hat sich gezeigt, dass *writer2latex* zu einem wichtigen Programm in meiner Sammlung von Konvertierungsprogrammen geworden ist. Für MS-Excel-Tabellen gibt es übrigens *Calc2LaTeX*, das aber nicht von mir getestet worden ist.

Literatur

- [1] Elke Niedermair und Michael Niedermair: *$\text{\LaTeX}$ – Das Praxisbuch*; Franzis-Verlag, Poing; 2005.

Komfortable L^AT_EX-Systeme zum Mitnehmen

Robert Winkler

Live CD/DVDs ermöglichen den Einsatz von L^AT_EX-Arbeitsumgebungen ohne die Notwendigkeit einer aufwändigen Installation. Im Artikel werden typische L^AT_EX-Editoren vorgestellt, sowie aktuelle Live CD/DVDs, die deren mobilen Einsatz ermöglichen.

Einleitung

Über die Vorteile, anspruchsvolle Dokumente in L^AT_EX zu setzen, anstatt sogenannte »Standardsoftware« wie Word und Star-/OpenOffice zu benutzen, brauche ich der geneigten Leserschaft der T_EXnischen Komödie wohl nicht zu referieren. Daher kann sich auch jeder die unangenehme Situation vorstellen, dass jemand sein Opus Magnum basierend auf L^AT_EX erstellen möchte, dies aber an der Institution, bei der man beschäftigt ist, nicht vorgesehen ist und daher die benötigte Software nicht installiert ist, bzw. meistens auch nicht installiert werden darf.

Einen Schritt in die richtige Richtung stellen die T_EX Live CD/DVDs dar, aber ich persönlich kann mich nicht mit den von Windows angebotenen und bezüglich L^AT_EX recht limitierten Editiermöglichkeiten anfreunden.

Dieser Artikel beschreibt daher einige Live-Systeme, die völlig ohne Festplatteninstallation auskommen und neben L^AT_EX auch eine komfortable Arbeitsumgebung mitbringen. Diese stellen auch im privaten Anwenderbereich eine Alternative für Menschen dar, die – ohne viel Aufwand zu betreiben – L^AT_EX einfach mal ausprobieren wollen.

Das Prinzip

CD einlegen – Nichtstun – Losarbeiten. Aus dieser Idee hat Klaus Knopper die CD-KNOPPIX [4], die »Mutter aller Live-CDs«, entwickelt. Grundsystem ist dabei eine bootfähige Linux-CD mit automatischer Hardwareerkennung. Darauf ist dann eine Arbeitsumgebung mit unterschiedlichen Programmen aufgebaut. Inzwischen existieren die unterschiedlichsten Varianten von

Live-CDs, die sich teilweise stark in den mitgelieferten Softwarepaketen unterscheiden. KNOPPIX selbst liefert derzeit kein L^AT_EX mehr mit, dafür sind andere in die Bresche gesprungen. Einige Distributionen werden gleich vorgestellt, zuvor jedoch ein paar Programme, die das Leben mit L^AT_EX schöner und einfacher machen.

Die Programme

Meine Liste ist sicher nicht vollständig, sollte aber für jeden Nutzertypus mindestens ein Programm dabeihaben. Grundsätzliches zur Problematik »L^AT_EX und WYSIWYG« (what-you-see-is-what-you-get) und ein Vergleich unterschiedlicher Konzepte findet sich in [7]. Üblicherweise befinden sich auf den Live-Systemen noch weitere Texteditoren, die zum T_EXen eingesetzt werden können, wie vi(m), die hier jedoch keine Erwähnung finden sollen.

Emacs/AUCT_EX

Vermutlich der einzige Editor, dessen Anhänger eine eigene Religion gegründet haben; die Church of Emacs. Zusammen mit der Erweiterung AUCT_EX entstand eine einzigartige Editierumgebung für T_EX/L^AT_EX.

T_EXmacs

Eine WYSIWYG-Umgebung für L^AT_EX; nach eigenen Angaben inspiriert von Emacs und T_EX.

Kile

Die einzige Editierumgebung, über die bisher noch nicht viel in der T_EXnischen Komödie zu lesen war. Kile steht für »KDE Integrated L^AT_EX Environment« und vereinigt so ziemlich alle Werkzeuge, die man zum Schreiben von L^AT_EX-Dokumenten braucht, unter einer Oberfläche. Die wesentlichen Ausstattungsmerkmale des Editors sind:

- Dokumenten-Wizard: zum schnellen Definieren der wichtigsten Eigenschaften eines neuen Dokuments
- Vorlagenverwaltung
- Syntax Highlighting

- Autokomplettierung von Befehlen und Umgebungen:
Bei `\begin{irgendwas}` wird gleich ein `\end{irgendwas}` eingefügt
- Strukturbrowser: Navigation im Text anhand der Dokumentenstruktur
- Fehlerbrowser
- Inverse und Vorwärtssuche `*.dvi` $\longleftrightarrow$ `*.tex`: Springen von einer Stelle im `*.dvi`-file zur entsprechenden Stelle im L^AT_EX-Text und umgekehrt.
Voraussetzung: `\usepackage[active]{srcltx}` in der Präambel.
- Eingabehilfen für Sonderzeichen und Tabellen
- Verwaltung von L^AT_EX-Projekten
- L^AT_EX Befehlsreferenz

Diverse Buttons erlauben das direkte Starten von Aktionen wie das Kompilieren zu `*.pdf` oder `*.dvi`, die Ansicht von erzeugten Dokumenten, Konvertierungen usw.

Neben dem eigentlichen Editierfenster gibt es dabei noch einen Rahmen auf der linken Seite, der entweder für Sonderzeichen, als Eingabehilfe für Metapostbefehle, als Strukturbrowser oder für Datei-/Projektverwaltung eingesetzt werden kann, sowie ein äußerst praktisches kleines Fenster unten, das wahlweise L^AT_EX-Meldungen ausgibt oder als Befehlsfenster benutzt werden kann. Nähere Informationen und Screenshots finden sich unter [3].

LyX

Bezeichnet sich selbst als WYSIWYM-Programm (what-you-see-is-what-you-mean), was auch den Kern der Sache ziemlich gut trifft. Zu diesem Editor finden sich mehrere Artikel in der T_EXnischen Komödie, auf die ich hiermit verweise.

Live Distributionen für L^AT_EX

Die hier vorgestellten Live-Distributionen stellen nur eine recht willkürliche Auswahl dar und beziehen sich auf den momentanen Status. Eine ausführliche Liste erhältlicher Distributionen mit Beschreibungen findet sich unter [1]. Da sich die Distributionen mit einer neuen Version bezüglich der mitgelieferten Programme ändern können, möchte ich dem potentiellen Nutzer generell

	KANOTIX	Quantian	Puppy
Emacs/AUCT $\text{\TeX}$		✓	
$\text{\TeX}$ macs		✓	
Kile	✓	✓	
L $\text{\Y}$ X	✓	✓	
Andere Editoren	✓	✓	✓
$\text{\LaTeX}$	✓	✓	$\text{\TeX}$ Live CD/DVD
Besonderheiten	Oft auf Heft- CD/ DVDs	Speziell für Wis- senschaftler	Multisession-CD möglich, extrem flott, leider α - Stadium

Tabelle 1: Live-Distributionen für $\text{\LaTeX}$

empfehlen, sich vor der Anschaffung die Paketlisten (engl. »package list«, üblicherweise auf der jeweiligen Distributions-Homepage zu finden) anzuschauen.

KANOTIX

KANOTIX [2] ist ein momentan recht populäres KNOPPIX-Derivat mit einer großen Vielfalt an Programmen. Mit dabei sind auch ein $\text{\LaTeX}$ -System und die beiden Arbeitsumgebungen Kile und L $\text{\Y}$ X. In einigen Linux-Zeitschriften können auch CD/DVDs mit KANOTIX gefunden werden, zudem besteht die Möglichkeit, die Distribution auf eine Festplatte zu installieren.

Quantian

Quantian [6], auch abgeleitet von KNOPPIX, ist vor allem für wissenschaftliche Anwender gedacht. Neben vielen zusätzlichen Programmen zur Lösung numerischer Probleme bietet es unter den vorgestellten Systemen auch die größte Auswahl an $\text{\LaTeX}$ -Anwendungen. Neben Emacs/AUCT $\text{\TeX}$ sind dies $\text{\TeX}$ macs, Kile und L $\text{\Y}$ X.

Puppy Multisession CD

Ein vollkommen neues Konzept verfolgt die Puppy Multisession CD [5]. Ich will gleich mal vorausschicken, dass es sich hier nicht um eine Distribution handelt, die speziell auf das Schreiben von L^AT_EX-Dokumenten ausgelegt ist. Aber das Konzept ist so interessant, dass es doch für den einen oder anderen eine interessante Alternative darstellen könnte. Diese Distribution hat die folgenden Eigenheiten:

- Das Gesamtsystem ist komprimiert etwa 50 bis 60 Mb groß, passt also auch auf einen USB-Stick o.ä.
- Alle Anwendungen werden beim Booten in den Arbeitsspeicher gespielt, so dass alle Programme extrem flott starten.
- Nach dem Starten kann die CD entnommen und durch eine T_EX Live CD/DVD ersetzt werden.
- Am Ende der Sitzung, kann man die System-CD wieder einlegen. Alle Änderungen bei Einstellungen und die bearbeiteten Daten werden wieder auf die CD zurückgeschrieben (Voraussetzung ist natürlich ein CD-Brenner).

Durch das Abspeichern der jeweils letzten Version hat man ein Backup jeder geänderten Datei auf CD. Die Sitzungen werden dabei in einen Track gespeichert, dessen Name aus Datum und Uhrzeit besteht. Auf diese Verzeichnisse kann man dann auch z. B. von Windows aus zugreifen. Werden Daten gelöscht, werden diese nicht mehr weitervererbt, sind aber nach wie in den älteren Tracks vorhanden.

Nochmal die Schritte einer Puppy-L^AT_EX-Sitzung in chronologischer Folge:

1. Booten von der Puppy-CD.
2. Austauschen der Puppy-CD gegen T_EX Live CD/DVD (wenn nur ein Laufwerk verfügbar ist).
3. Vor dem Herunterfahren: Einlegen der Puppy-CD und Brennen der Daten (automatisch).

Übrigens: Sollte die CD voll sein, wird man aufgefordert, eine leere CD einzulegen. Dann wird diese mit dem System und der aktuellen Sitzung neu erstellt und weiter geht's.

Der Status dieses Projektes ist Alpha und hat noch einige Verbesserungen nötig. Aber es sind beste Ansätze für ein immer-dabei $\text{\LaTeX}$ -Live-System vorhanden. Besonders für unerschrockene Bastler, die sich ihre eigene $\text{\LaTeX}$ Live CD bauen wollen, stellt dieses System eine äußerst innovative Basis dar.

Und wo speichere ich meine Dokumente?

Im Falle von Puppy hat sich die Frage schon erübrigt. Bei den anderen Varianten kann auf beispielsweise auf Festplatte, Diskette oder – moderner – auf USB-Stick gespeichert werden. Damit sind auch Software und Daten sauber voneinander getrennt und ein Update besteht aus der Neuanschaffung einer CD.

Bezugsmöglichkeiten und Preise

Auf den Homepages der Distributionen finden sich Links zu den ISO-images, die man sich dann selbst auf CD oder DVD brennen kann. Alternativ kann man sich die Datenträger auch fertig gebrannt und bedruckt bestellen. Zu empfehlen ist beispielsweise der Service von <http://www.ISO4YOU.de>; dort sind alle oben genannten Distributionen für Preise ab 1,95€ plus Versand erhältlich.

Zusammenfassung

Ein mit »Standardsoftware« konfiguriertes System, das einem vor die Nase gesetzt wird, ist keine Entschuldigung mehr, kein $\text{\LaTeX}$ zu benutzen. Live-Distributionen lassen sich von (fast) jedem Rechner booten und ermöglichen sofortiges Arbeiten mit komfortablen Arbeitsumgebungen ohne lästige Konfiguration. Die Einstiegskosten sind äußerst moderat, zudem ist eine Entkoppelung von Software und Daten möglich, wodurch eventuelle Upgrades einfach und sicher durchzuführen sind. Das Beispiel für eine innovative Live Multisession-CD zeigt, dass sogar ein Sicherungs- und Backupmechanismus in den Sitzungsverlauf eingebaut sein kann.

Literatur und Software

- [1] *Distrowatch Homepage*; <http://distrowatch.com/>.
- [2] *KANOTIX Homepage*; <http://kanotix.com/info/index.php?lang=de>.
- [3] *Kile Homepage*; <http://kile.sourceforge.net>.
- [4] *KNOPPIX Homepage*; <http://www.knoppix.org/>.
- [5] *Puppy Homepage*; <http://www.goosee.com/puppy/>.
- [6] *Quantian Homepage*; <http://dirk.eddelbuettel.com/quantian.html>.
- [7] D. Kastrup: *$\LaTeX$ und WYSIWYG? - preview-latex unter Emacs und andere Ansätze; Die $\TeX$ nische Komödie*; 4, S. 10–26; 2004.

T_EX-Beiprogramm

Neue Pakete auf CTAN

Jürgen Fenn

Der Beitrag stellt neue Pakete auf CTAN seit Mitte April 2005 bis zum Redaktionsschluss vor (letzte Änderung am 17. Juli 2005). Die Liste folgt der umgekehrten chronologischen Reihenfolge.

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

microimp von der Firma *MicroPress* ist eine kostenlose (aber keine freie, weil nicht *open source*) Distribution einer neuen L^AT_EX-basierten Textverarbeitung für 32-Bit MS Windows. Es handelt sich um den Release 0.12, der noch nicht völlig fertig gestellt ist. Er soll aber bereits recht gut funktionieren.

MicroIMP ist eine intuitiv bedienbare, typische WYSIWYG-Oberfläche, die L^yX Konkurrenz machen könnte. Man muss kein T_EX- oder L^AT_EX-Markup lernen, sondern kann den Text direkt eingeben und über die Menüs formatieren. Dem *Readme* zufolge funktioniert das Programm sofort nach der Installation »out of the box« und bietet neben anderen Features auch mathematischen Formelsatz, Silbentrennung und Ligaturen während der Eingabe. Es gibt auch eine Rechtschreibprüfung. Bisher werden Deutsch, Englisch, Französisch, Niederländisch und Italienisch unterstützt. Es existiert ein RTF-Importfilter.

Weitere Features können später ergänzt werden, wenn sie zur Verfügung stehen.

Daneben können T_EX-Befehle auch direkt eingegeben werden.

CTAN:/systems/win32/microimp

de-macro ist ein Python-Skript, mit dem man eigene Makros konvertieren kann, beispielsweise um ein L^AT_EX-Dokument weiterzugeben, das man veröffentlichen oder mit anderen austauschen möchte.

CTAN:/support/de-macro

commath von Wolfgang Putschögl stellt Befehle zur flexiblen Formatierung von mathematischen Formeln zur Verfügung.

CTAN:/macros/latex/contrib/commath

iopart-num ist ein B^IB^TE^X-Style für die Zeitschriften des *Institute of Physics* (IOP) einschließlich des *Journal of Physics*. Es wird ein »Harvard-artiger« Zitierstil erzeugt.

Autor ist *Mark Caprio*.

CTAN:/biblio/bibtex/contrib/iopart-num

mnsymbol von *Achim Blumensath* ist ein Symbol-Font, der zusammen mit der Schrift *Adobe MinionPro* verwendet werden kann. Es handelt sich um eine Betaversion.

CTAN:/fonts/MnSymbol

makecell stellt eine Reihe von Befehlen zur Auszeichnung und zur Gestaltung von Tabellen bereit.

Das Paket baut auf einspaltigem Tabellenlayout auf und sorgt für ein einheitliches Erscheinungsbild der Kopfzeilen aller Tabellen in einem Dokument. Das Paket stellt auch Befehle zur Erzeugung von Zellen bereit, die mehrfach umrandet sind, sowie ein Makro, mit dem man das vertikale Padding in allen Zellen einer Tabelle ändern kann (ähnlich dem Paket *tbls*). Auch mehrspaltige Zellen, die das Paket *multirow* verwenden, werden unterstützt.

Autorin des Pakets ist *Olga Lapko*.

CTAN:/macros/latex/contrib/makecell

tex2tok ist ein Konverter, der T_EX-Quellcode in Tokens umsetzen kann. Autor ist *Jonathan Fine*.

CTAN:/support/tex2tok

LFB von *Laurie Field* ist eine griechische Schriftart, die in METAFONT geschrieben wurde. Bisher gibt es normale und fette Schnitte. Die Fonts sind der Schrift *Bodoni* nachempfunden, die der Paketautorin in alten Büchern in der Bibliothek ihrer Schule aufgefallen war. Stilistisch wirkt diese Schrift vielleicht etwas »exotisch«, verglichen mit den griechischen Fonts, die allgemein verwendet werden. Sie sollte modern anmuten und gut zu *Computer Modern* passen. Ein Ligatur-Schema für Altgriechisch ist beigelegt.

CTAN:/fonts/greek/lfb

byzfonts ist eine neue Schriftart, mit der man byzantinische Kirchenmusik setzen kann, die in der griechisch-orthodoxen Kirche gespielt wird. Die Fonts stehen unter der LGPL.

Die Autoren sind *Ioannis A. Vamvakas* und *Panagiotis Kotopoulos*.

CTAN:/fonts/byzfonts

sudoku stellt eine Umgebung bereit, um Sudoku-Puzzle-Raster zu erzeugen.

Ein Sudoku-Puzzle ist ein Gitter aus 9×9 Feldern. Einige der Felder enthalten Zahlen. Die Regeln sind einfach: Jede Spalte und jede Zeile muss Zahlen von 1 bis 9 enthalten, und jede Box aus 3×3 Feldern ebenfalls. Jede Zahl darf nur jeweils einmal vorkommen. Mehr erfährt man unter <http://www.sudoku.org.uk>.

Autor des Pakets ist *Paul Abraham*.

CTAN:/macros/latex/contrib/sudoku

aramaic und *nabatean* sind zwei Bundles mit historischen Schriftarten, die *Peter Wilson* entwickelt hat. Alle Schriften aus seiner Serie (sie umfasst außerdem die Pakete *linearb*, *oldprsn* und *ugarite*) stehen jetzt neu als *Type 1*-Fonts und METAFONT-Quellen zur Verfügung.

CTAN:/fonts/archaic/aramaic

CTAN:/fonts/archaic/nabatean

AAS_TeX ist ein neuer Style für Aufsätze in den Zeitschriften der *American Astronomical Society*, den *Sara Zimmerman* entwickelt hat.

CTAN:/macros/latex/contrib/aastex

classlist von *Heiko Oberdiek* geht auf eine Diskussion in der Newsgroup `comp.text.tex` zurück. Das Paket wird noch vor dem Laden der Dokumentenklasse mit dem Befehl `\RequirePackage{classlist}` eingebunden und speichert die Klasse und alle verwendeten Optionen in einer Liste, die auf dem Bildschirm ausgegeben werden kann.

CTAN:/macros/latex/contrib/oberdiek

perception ist ein $\text{\LaTeX}$ -Style für die Zeitschrift *Perception*, den *Jed Singer* bereitstellt.

CTAN:/biblio/bibtex/contrib/perception

sides ist eine $\text{\LaTeX}$ -Klasse zum Setzen von Theaterstücken, die auf der Klasse *plari* von *Antti-Juhani Kaijanaho* aus dem Jahr 1998 beruht. *Wing L. Mui* hat diese Klasse weiterentwickelt und eine Reihe von Änderungen bei der Formatierung eingearbeitet. Insbesondere wurde das Problem mit Waisenkindern beseitigt.

CTAN:/macros/latex/contrib/sides

fig2vect konvertiert Graphikdateien in den Formaten *jFig*, *XFig* und *WinFig* in Vektorgraphiken (*METAPOST*, *EPS*, *PDF*, $\text{\TeX}$ und *SVG*). Das Projekt ist auf <http://fig2vect.sourceforge.net> zu Hause.

Autor ist *Dirk Krause*.

CTAN:/support/fig2vect

koma-script-SFS ist eine *clo*-Datei, um mit der Briefklasse *scrlltr2* aus dem *KOMA-Script*-Bundle von *Markus Kohm* finnische Briefe zu setzen. Autor und Maintainer des Pakets ist *Hannu Väisänen*.

CTAN:/macros/latex/contrib/koma-script-SFS

cclicenses stellt Makros bereit, mit denen man die Logos für *Creative Commons*-Lizenzen 1.0 und 2.0 erzeugen kann. Autor ist *Gianluca Pignalberi*.

CTAN:/macros/latex/contrib/cclicenses

xpdfopen stellt die zwei kleinen Programme *pdfopen* und *pdfclose* zur Verfügung, um *Acrobat Reader* 5.0.10 und 7.0 für *Linux* entweder von der Kommandozeile oder aus einem Shell-Skript heraus zu steuern.

CTAN:/support/xpdfopen

truelypmetrics enthält alles, was man braucht, um die *TrueType*-Schriften *Monotype Berling*, *Linotype Frutiger* und *Linotype Palatino* mit L^AT_EX verwenden zu können (außer den Fonts selbst, versteht sich. . .).

Paketautor ist *Marc Penninga*.

CTAN:/fonts/truetypermetrics

LEd ist ein neuer freier L^AT_EX-Editor für *MS Windows*, der an die graphische Oberfläche von *T_EXnicCenter* erinnert.

Die Entwickler sind *Sebastian Deorowicz* und *Adam Skorczynski*, und das Projekt ist unter der URL www.latexeditor.org zu erreichen.

CTAN:/support/LEd

sketch von *Eugene K. Ressler* ist ein kleines und einfaches System zur Erzeugung von Zeichnungen dreidimensionaler Objekte und Bildern. Es war ursprünglich entwickelt worden, um ein Lehrbuch zu illustrieren, nachdem sich kein brauchbares Werkzeug dafür finden ließ. Zeichen-Tools mit graphischer Oberfläche und photorealistischer Darstellung hatten sich für den Entwickler zur Erstellung schöner mathematischer Zeichnungen ohne allzu feine Einzelheiten als unbrauchbar erwiesen. Die Zeichnungen sollten auch leicht mit T_EX verwendbar sein. Die Eingabesprache erinnert an PSTricks, so dass Anwender, die an PSTricks gewöhnt sind, sich leicht einarbeiten können. Weitere Informationen findet man unter <http://www.frontiernet.net/~eugene.ressler>

CTAN:/graphics/sketch

fontools will die Verwendung von Schriften, vor allem von *Truetype*- und *Opentype*-Schriften mit L^AT_EX und **fontinst** vereinfachen. Dazu werden eine Reihe von Tools zur Verfügung gestellt.

Autor ist *Marc Penninga*.

CTAN:/fonts/utilities/fontools

Zigaretenschachtelhuellenzeichnung ist eine Idee von *Régis Lachaume*. Das Paket zeichnet Hüllen, mit denen die Warnhinweise auf Zigaretenschachteln verhüllt werden können.

CTAN:/graphics/zigaretten

getfileversion von *Harald Harders* ist ein einfaches Shell-Skript, das die Versionsnummer und das Datum einer $\text{\LaTeX}$ -Klasse oder eines $\text{\LaTeX}$ -Styles ausgibt.

CTAN:/support/getfileversion

harmony von *Arnim Wegner* ist ein Paket zum Setzen von Harmoniesymbolen in musikwissenschaftlichen Texten.

CTAN:/macros/latex/contrib/harmony

xlop steht für »*eXtra Large OPeration*«. Mit dem Paket von *Jean-Côme Charpentier* kann man Rechenwege schriftlich »wie in der Schule« oder »inline« darstellen.

CTAN:/macros/generic/xlop

maltese erleichtert die Eingabe von Zeichen außerhalb der Latin-Zeichensätze in maltesischer Sprache in $\text{\LaTeX}$. Autor ist *Paulseph-John Farrugia*.

CTAN:/language/maltese

thmbox definiert eine Umgebung, die einen Rahmen oder ähnliche Dekorationen rund um Theoreme, Definitionen und ähnliche Objekte erzeugen kann. Dabei wird das Standardmakro `\newtheorem` umdefiniert. Autor des Paketes ist *Emmanuel Beffara*.

CTAN:/macros/latex/contrib/thmbox

User's Guide for AMS package (vietnamese version) ist, wie der Name schon sagt, die vietnamesische Übersetzung der Anleitung zu $\text{\LaTeX}$. Der Autor (oder die Autorin) heißt *Kyanh*.

CTAN:/info/amslatex/vietnamese

tamefloats ist der neue Name des Pakets *noreinserts* von *Uwe Lück*, das in der letzten Ausgabe der DTK an dieser Stelle vorgestellt worden war.

CTAN:/macros/latex/contrib/tamefloats

bsheaders von *Maurizio Loretì* ändert die Formatierung der Überschriften: Sie werden in serifenlosen Schriften dargestellt und oben sowie unten von Linien eingeschlossen, die über die ganze Textbreite gehen.

CTAN:/macros/latex/contrib/bsheaders

Wettbewerb – Gestaltung von Aufklebern

Tobias Sterzl

Liebe Mitglieder,

allen dürften die kleinen Aufkleber bekannt sein, die heute auf vielen Computern und Notebooks zu sehen sind. Oft weisen diese Aufkleber darauf hin, dass in den Computer spezielle Hardwarekomponenten eingebaut sind. Auch das Logo eines favorisierten Betriebssystems wird gern aufgeklebt. Hier wäre es doch schön, wenn die $\text{\TeX}$ -, $\text{\LaTeX}$ - oder $\text{\ConTeXt}$ -Benutzer ihre Rechner ebenfalls mit einem solchen Aufkleber schmücken könnten, der den interessierten Computernutzern mitteilt, dass die Texte *Formatted by $\text{\TeX}$* sind.

Eure Kreativität ist gefordert, einen oder mehrere Aufkleber in einer bis drei verschiedenen Größen zu gestalten:

1. ca. 20×30 mm, wie die kleinen »Intel inside«-Aufkleber
2. ca. 50×100 mm für Koffer, Taschen usw.
3. ca. 300×500 mm für Hinweisschilder, Autos ...

Die Maße sollen nur ungefähr die Größenordnung angeben (es darf davon abgewichen werden, hochkant, quer, oval, dreieckig o. ä. sein). Die Verwendung von Farben ist erwünscht.

Das DANTE e.V.-Logo¹ darf verwendet werden, muss aber nicht – es geht nicht vorrangig um einen Aufkleber für oder von DANTE e.V., sondern für $\text{\TeX}$ & Co. (www.dante.de als Info-Site darf vorkommen).

Einsendeschluss ist der 30. November 2005

Die Vorschläge sind an das Büro von DANTE e.V. in Heidelberg zu senden (Post, E-Mail). Pro Person dürfen mehrere Vorschläge pro Kategorie gemacht werden. DANTE e.V. wird die Siegerentwürfe produzieren. Die Nutzungsrechte gehen an DANTE e.V.

Jede Kategorie wird einzeln bewertet und auf die Gewinner warten attraktive Preise!

...na dann, nichts wie ran an die Buntstifte ...

¹kann unter www.dante.de/wettbewerb heruntergeladen werden

Spielplan

Termine

- 23.–25. 8. 2005** TUG 2005
Wuhan, China
<http://www.rons.net.cn/tug2005.html>
- 29.–30. 9. 2005** Herbsttagung und 33. Mitgliederversammlung von DANTE e.V.
Universität Kiel, Schleswig-Holstein
<http://www.dante.de/dante/events/mv33/>
- 8.–10. 3. 2006** DANTE 2006 – T_EX-Tagung und 34. Mitgliederversammlung von DANTE e.V.
Freie Universität Berlin
<http://www.dante.de/dante/events/dante2006/>



Stammtische

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

Aachen

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

Berlin

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

Bremen

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

Darmstadt

Karlheinz Geyer
karlheinz.geyer@LHSystems.com
Restaurant Poseidon
Rheinstraße 41
64283 Darmstadt
Erster Freitag im Monat, ab 19.30 Uhr

Dresden

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

Düsseldorf

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

Erlangen

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

Freiburg

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

Hannover

Mark Heisterkamp
heisterkamp@rrzn.uni-hannover.de
Seminarraum RRZN
Schloßwender Straße 5
Zweiter Mittwoch von geraden Monaten,
18.30 Uhr

Heidelberg

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

Karlsruhe

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

Kiel

Karsten Heymann
karsten.heyman@gmx.de
Letzter Donnerstag im Monat, 19.00 Uhr
Ort siehe <http://zaubrerer.net/latexwiki>

Köln

Bruno Hopp
b.hopp@lepkes-frings.de
Institut für Kristallographie
Zülpicher Straße 49b
Letzter Mittwoch im Monat, 19.30 Uhr

München

Michael Niedermair
m.g.n@gmx.de
Wirtshaus »Löwe am Markt«
Dreifaltigkeitsplatz 4
Erster Dienstag im Monat, 19.00 Uhr

Münster

Johannes Reese
reese@linguist.de
Gaststätte »Sabroso«
Mauritzstraße 19
nach Vereinbarung

Stuttgart

Bernd Raichle
bernd.raichle@gmx.de
Bar e Ristorante »Valle«

Geschwister-Scholl-Str. 3
Zweiter Dienstag im Monat, 19.30 Uhr

Trier

Peter Schuster
schu5107@uni-trier.de
Fetzenkneipe (Haus Fetzenreich)
Sichelstraße 36 (beim Sieh-Um-Dich)
54290 Trier
Dritter Montag des Monats, 20.15 Uhr

Wuppertal

Andreas Schrell
Tel.: 0202/50 63 81
schrell@wupperonline.de
Restaurant Croatia »Haus Johannisberg«
Südstraße 10
an der Schwimmooper Wuppertal-Elberfeld
Zweiter Donnerstag im Monat, 19.30 Uhr

Zürich

Johannes Reese
reese@spw.unizh.ch
nach Vereinbarung



E.Ruff; G.Mannigel



U.Siart; M.Niedermair



V.Holzhauser; U.Siart



U.Dirr; S.Schellshorn

Der Münchener Stammtisch (Fotos: Uwe Siart)

Adressen

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

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

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

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

Präsidium

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

Server

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

Autoren/Organisatoren

Hendri Adriaens Warandelaan 2 Postbus 90153 NL-5000 LE Tilburg hendri@uvt.nl	[47]	Michael Niedermair Am Malerwinkel 16 85778 Haimhausen	[27, 58]
Jürgen Fenn Friedensallee 174/20 63263 Neu-Isenburg juergen.fenn@gmx.de	[72]	Erich Ruff Kruckenburgstr. 21 81375 München Erich_Ruff@T-Online.de	[8]
Ulrike Fischer Kastanienstraße 2 53721 Siegburg skak@nililand.de	[31]	Volker RW Schaa siehe Seite 82	[4, 6]
Karsten Heymann Hasseldieksdammer Weg 41 24114 Kiel karsten.heyman@gmx.de	[6]	Peter Seitz Waldaustraße 13 90441 Nürnberg	[21]
Klaus Höppner siehe Seite 82	[4]	Tobias Sterzl siehe Seite 82	[77]
Uwe Kern Lessingstraße 14 65189 Wiesbaden u.kern@web.de	[47]	Herbert Voß Wasgenstr. 21 14129 Berlin hvoss@tug.org	[3]
		Robert Winkler Fromannstraße 5 07743 Jena Robert.Winkler@bioprocess.org	[65]

Die T_EXnische Komödie

17. Jahrgang Heft 3/2005 August 2005

Impressum

Editorial

Hinter der Bühne

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

T_EX-Theatertage

- 8 2²² – DANTE e.V. wird sechzehn
- 21 Die EuroT_EX – oder wieso der 16-te ein *runder* Geburtstag ist

Bretter, die die Welt bedeuten

- 27 Neues von ϵ_X T_EX: TFM-Daten anschaulich darstellen
- 31 Schach mit `skak.sty`
- 47 Neue Mechanismen in der Verarbeitung von Keys
- 59 *writer2latex*: Eine Praxiserfahrung
- 65 Komfortable L^AT_EX-Systeme zum Mitnehmen

T_EX-Beiprogramm

- 72 Neue Pakete auf CTAN
- 78 Wettbewerb – Gestaltung von Aufklebern

Spielplan

- 79 Termine
- 80 Stammtische

Adressen

- 83 Autoren/Organisatoren