

Die T_EXnische Komödie

dante

Deutschsprachige
Anwendervereinigung T_EX e.V.

23. Jahrgang Heft 3/2011 August 2011

3/2011

Impressum

»Die \TeX nische Komödie« ist die Mitgliedszeitschrift von DANTE e.V. Der Bezugspreis ist im Mitgliedsbeitrag enthalten. Namentlich gekennzeichnete Beiträge geben die Meinung der Autoren wieder. Reproduktion oder Nutzung der erschienenen Beiträge durch konventionelle, elektronische oder beliebige andere Verfahren ist nicht gestattet. Alle Rechte zur weiteren Verwendung außerhalb von DANTE e.V. liegen bei den jeweiligen Autoren.

Beiträge sollten in Standard- \LaTeX -Quellcode unter Verwendung der Dokumentenklasse dtk erstellt und per E-Mail oder Datenträger (CD) an untenstehende Adresse der Redaktion geschickt werden. Sind spezielle Makros, \LaTeX -Pakete oder Schriften notwendig, so müssen auch diese komplett mitgeliefert werden. Außerdem müssen sie auf Anfrage Interessierten zugänglich gemacht werden. Weitere Informationen für Autoren findet man auf der Projektseite <http://projekte.dante.de/DTK/AutorInfo> von DANTE e.V.

Diese Ausgabe wurde mit LuaTeX, Version beta-0.70.1-2011061410 (rev 4277) (TeX Live 2011) erstellt. Als Standard-Schriften kamen die Fonts \TeX Gyre Pagella, \TeX Gyre Heros, Bera Mono und Latin Modern Math zum Einsatz.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 2700

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

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

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

Redaktion: Herbert Voß (verantwortlicher Redakteur)

Mitarbeit: Rudolf Herrmann Gert Ingold Jürgen Lübeck
Rolf Niepraschk Heiko Oberdiek Christine Römer
Gert Seidl

Redaktionsschluss für Heft 4/2011: 15. Oktober 2011

ISSN 1434-5897

Editorial

Liebe Leserinnen und Leser,

wenn mich meine Studenten fragen, warum man die Mängel, die \TeX unzweifelhaft aufweist, nicht einfach behebt, indem man es in einer »modernen« Programmiersprache neu schreibt, antworte ich immer, dass sich »bislang jeder daran verhalten hat«. Ob diese Aussage auch noch für die aktuellen Neuentwicklungen wie \XeTeX oder \LuaTeX gilt, wird sich zeigen müssen. Immerhin sind dies zwei vielversprechende und praktisch anwendbare Varianten von \TeX . Mit \XeTeX scheitert man beispielsweise an eigentlich trivialen Dingen, wie dem Clipping bei einer Grafik (siehe meinen Beitrag auf Seite 68). Bei \LuaTeX ist es immer noch die Geschwindigkeit, die bei sehr großen Dokumenten negativ auffällt. Da nützen auch die unzweifelhaft vorhandenen Vorteile bei der Einbindung von OpenType- oder TrueType-Schriften wenig. Diese Ausgabe von »Die \TeX nische Komödie« wurde wegen der beiden Artikel zum Thema $\Lua\LaTeX$, die Anwendungen zum Einsatz von Lua zeigen, der Einfachheit halber ebenfalls mit $\Lua\LaTeX$ gesetzt. Dabei galt es in einigen Fällen mit einem »segmentation fault« (Speicherzugriffsproblem) zu kämpfen, was Anwendern selten Freude bereitet. Da die Entwicklung von \XeTeX praktisch eingestellt wurde, bleibt die Hoffnung, dass \LuaTeX noch weitere Verbesserungen erfahren wird und wirklich einmal der Nachfolger von \pdfTeX werden kann. Immer aber bleibt \TeX im Mittelpunkt, was sein Entwickler wohl damals auch nicht erwartet hätte. Fan Chung Graham hat als Mathematikerin und Hobbymalerin alle ihr wichtigen Köpfe der Mathematik, die ihr bisher in ihrem beruflichen Leben begegnet sind, in Aquarellen verewigt: <http://www.math.ucsd.edu/~fan/paint/math.html>. Donald Knuth zählt auch dazu (siehe Seite 5).

Im letzten Heft wurden die Möglichkeiten der Einbindung der Skriptsprache Lua für die einfache Durchführung arithmetischer Berechnungen aufgezeigt. In dieser Ausgabe finden Sie weitere Anwendungen von Lua. Zum einen, wie man eine tabellarische Anordnung der Zeichen der neuen mathematischen Schrift Latin Modern Math erhält, und zum anderen, wie man eine Musterliste aller Schriften einer bestimmten Familie ausgeben kann. Dies kann beispielsweise bei Latin Modern hilfreich sein, denn sie enthält eine Vielzahl an möglichen Varianten, die nicht immer bekannt sind.

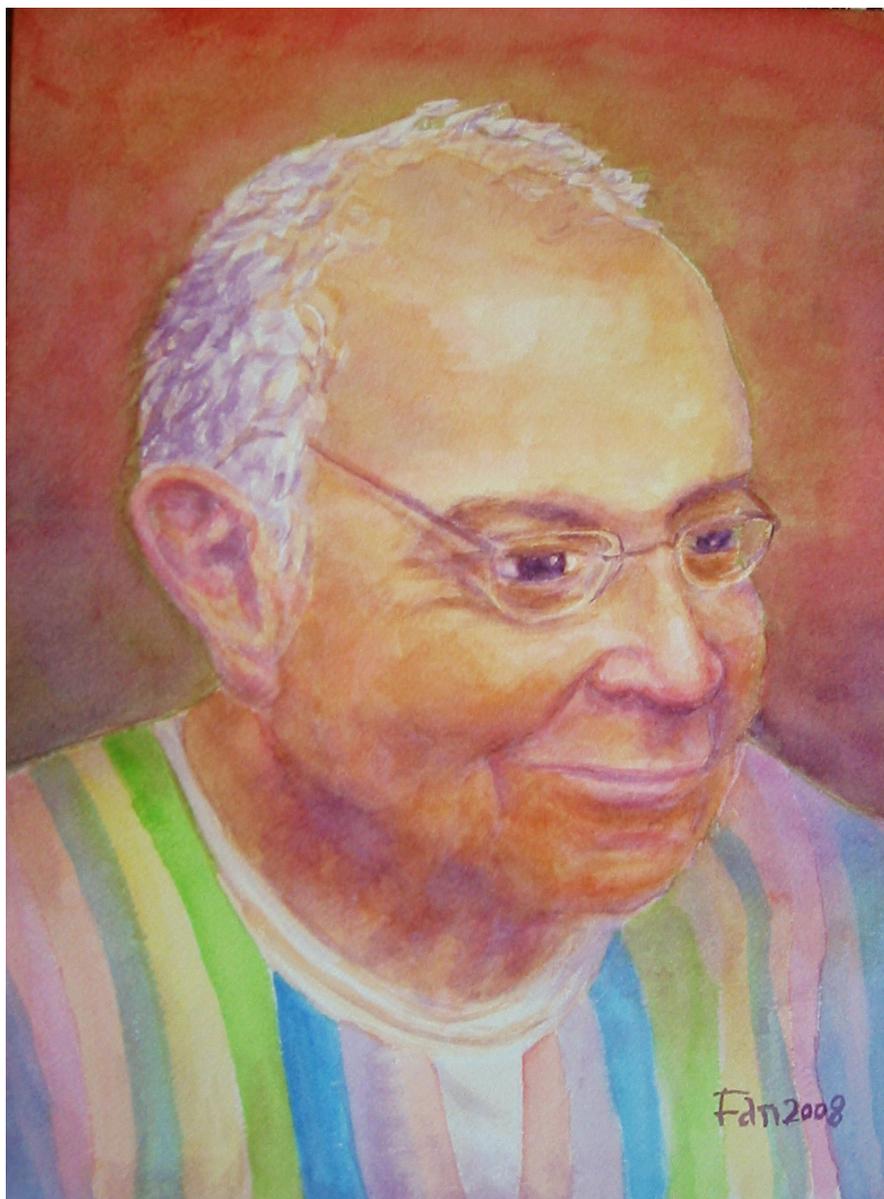
Weiterhin finden Sie in diesem Heft einen Artikel von Dominik Wagenführ zu den Grundlagen der Makrodefinition, die das Key/Value-Prinzip unterstüt-

zen, wie es mittlerweile von fast allen neuentwickelten Paketen angewendet wird. Den Artikel von Enrico Gregorio zur Installation von T_EX Live unter Ubuntu und anderen Linux-Systemen finden Sie als Nachdruck aus der TUGboat in deutscher Übersetzung. Reinhard Kotucha beschreibt die Funktionsweise des nicht mehr in T_EX Live vorhandenen Skripts `getnonfreefonts`, welches einige frei verwendbare, aber nicht freie Schriften installiert. Die Arbeitsgruppe der »Trennmuster« (<http://projekte.dante.de/Trennmuster/WebHome>) hat eine neue Version erstellt, deren Einbindung von Rolf Niepraschk in einem Artikel gezeigt wird. Axel Kielhorn gibt einen Ausblick in die Zukunft der elektronischen Bücher, die ein zukunftssträchtiges Anwendungsgebiet für den Einsatz von L^AT_EX sein können. Dass das Thema E-Book sehr kontrovers angegangen wird, kann man in Richard Stallmans Beitrag »The Danger of E-Books« nachlesen (<http://www.gnu.org/p/the-danger-of-ebooks.html>); die Eigentumsrechte beim gedruckten Buch sind eindeutig, beim E-Book wohl noch nicht.

Die aktuelle T_EX Live 2011 als DVD werden Sie voraussichtlich mit der nächsten Ausgabe von »Die T_EXnische Komödie« erhalten. MacT_EX-User sollten in jedem Fall <http://www.tug.org/mactex/2011/prefpane.html> beachten.

Ich wünsche Ihnen wie immer viel Spaß beim Lesen und verbleibe mit T_EXnischen Grüßen,

Ihr Herbert Voß



Aquarell von Fan Chung Graham: *Der Meister der Perfektion – ein Porträt von Don Knuth*

Hinter der Bühne

Vereinsinternes

Grußwort

Liebe Mitglieder,

während sich das Jahr 2011 offensichtlich durch die Verlegung von internationalen \TeX -Konferenzen auszeichnet: TUG 2011 von Kairo/Ägypten nach Trivandrum/Indien (19.–21. Oktober) und das 5. Internationale Con \TeX t-Treffen vom Geheimtipp vor St. Tropez auf der Insel Porquerolles/Frankreich nach Bassenge-Boirs/Belgien (19.–24. September), können wir für die Herbst-Tagung von DANTE e.V. versprechen, dass sie im hohen Süden in Garmisch-Partenkirchen stattfinden wird. »Hoher Süden« deshalb, weil Garmisch-Partenkirchen sowohl der bisher höchste Tagungsort (708 m ü. NN) von DANTE e.V. als auch der südlichste sein wird. Andreas Hirsch und Stephan Lukasczyk vom Werdenfels-Gymnasium organisieren die Tagung und laden ein mit dem Versprechen eines reizvollen touristischen Begleitprogrammes. Im Heft finden Sie die Einladung und die Bitte um Angebote zu Tutorien und Vorträgen.

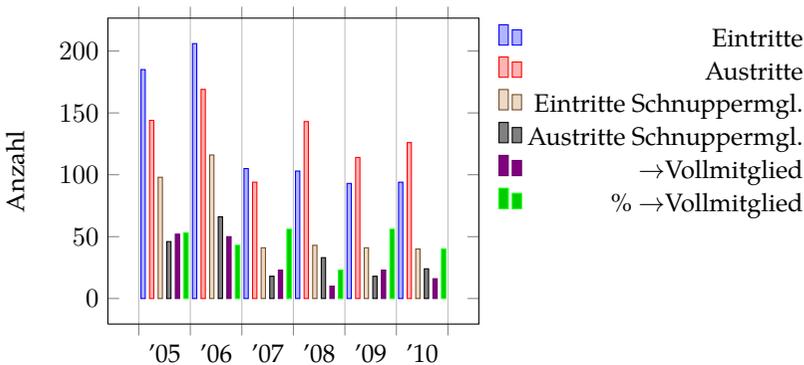
Eine Bemerkung im Bericht des Vorstands bei der 44. Mitgliederversammlung in Bremen, die in dieser Form auch Eingang in das in der letzten Ausgabe von »Die \TeX nische Komödie« abgedruckte Protokoll fand, hat zu Recht Irritationen bei unserem Partner »Lehmans Fachbuchhandlung GmbH« hervorgerufen. Über die Zusammenarbeit mit Lehmanns für die Produktion der \TeX Collection 2010 schrieben wir, dass es diesmal »nicht besonders gut« geklappt hätte.

Diese Bemerkung ist ohne weitere Erklärung missverständlich und war der Tatsache geschuldet, dass die jahrelang eingespielte Zusammenarbeit mit Lehmanns in der Person von Christoph Kaeder zu einem blinden Verstehen geführt hatte, alles kam aus einer Hand und wurde direkt erledigt. Nachdem er nun bei Lehmanns den Verantwortungsbereich »WebShop Manager« übernommen hat und nicht mehr für die Zusammenarbeit mit DANTE e.V. zuständig ist, wurden die Aufgaben auf verschiedene Schultern verteilt. Dadurch war es für uns nicht immer klar, wer für die einzelnen Bearbeitungsschritte zuständig war, und der vorher eingespielte Workflow erschien deutlich zäher. Wir haben aber erkannt, dass es sich hier um die Anlaufphase handelt, in der sich die Abläufe erst einspielen müssen.

Deshalb möchten wir an dieser Stelle verspätet die Gelegenheit nutzen, uns ganz herzlich bei Christoph Kaeder für die ausgezeichnete Zusammenarbeit zu bedanken. Wir glauben daran, dass wir mit Herrn Bönisch und seinem Team in gleicher Weise zukünftig zusammenarbeiten werden.

In den Monaten Juni und Juli herrschte ein reger E-Mail-Verkehr auf der vereinsinternen Liste `dante-ev` zu Themen wie »Argumente für DANTE-Beitritt«, »Dante–neue Liste für TeX-nische Fragen« und grundsätzliche Fragen zum Sinn und Zweck von Listen, Foren, Wikis. Viele der Vorschläge in den Diskussionsbeiträgen sind nur realisierbar, wenn es mehr ehrenamtliche Aktive für die Pflege dieser Dienste gäbe, die nicht nur ein kurzzeitiges Interesse haben, sondern auch bereit wären, sich über einen längeren Zeitraum zu engagieren.

Zu der konkreten Frage über Mitgliederbewegungen und der Attraktivität der Schnuppermitgliedschaften hat Karin Dornacher im Büro von DANTE e.V. die Daten zusammengestellt, die wir hier mit dem Fokus auf Veränderungen der Verhältnisse in komprimierter Form veröffentlichen.



Das Interesse an einer Schnuppermitgliedschaft lag nach anfänglich hohen Zahlen (≈ 100) in den letzten 4 Jahren bei nur noch 40 Anträgen. Im Durchschnitt lag der Prozentsatz der Schnuppermitglieder, die sich nach einem Jahr zu einer Vollmitgliedschaft entschlossen, bei 45 %.

Zu dem Vorschlag, die vollständigen Hefte von »Die TeXnische Komödie« früher zum allgemeinen Download zur Verfügung zu stellen, liegt inzwischen ein Antrag für die MV im Frühjahr 2012 vor, über den die Mitgliederversammlung beschließen wird.

Mit freundlichen Grüßen,

Volker RW Schaa Adelheid Grob

Vorsitzender Stellvertretende Vorsitzende

Einladung zur Herbsttagung und 45. Mitgliederversammlung von DANTE e.V.

Volker RW Schaa, Andreas Hirsch

Liebe Mitglieder von DANTE,

hiermit laden wir Sie herzlich zur Herbsttagung und 45. Mitgliederversammlung von DANTE e.V. ein. Zum ersten Mal werden wir in Garmisch-Partenkirchen zu Gast sein, dies ist nicht nur der bisher höchste Tagungsort von DANTE e.V., sondern auch der südlichste.

Die Tagung wird vom Werdenfels-Gymnasium ausgerichtet und findet am Samstag, den 1. Oktober 2011 am

Werdenfels-Gymnasium
E001
Wettersteinstraße 30
82467 Garmisch-Partenkirchen

statt.

Für Tagung und Mitgliederversammlung ist folgender Zeitplan vorgesehen:

Freitag, 30. September, ab 19:00 Uhr: Vorabendtreff
Samstag, 1. Oktober, 9:00 Uhr: Mitgliederversammlung
von DANTE e.V.
bis ca. 17:30 Uhr: Tagungsprogramm mit Vorträgen
ab 19:00 Uhr: Abendtreff
Sonntag, 2. Oktober, ab 10:00 Uhr: Touristikprogramm

Die Tagesordnung der Mitgliederversammlung lautet:

1. Begrüßung; Vorstellung des Vorstands; Tagesordnung
2. Bericht des Vorstands
3. Verschiedenes

Wie üblich sind auch Nichtmitglieder als Gäste herzlich willkommen.

Weitere Informationen zur Tagung finden Sie unter <http://www.dante.de/events/mv45.html>. Bitte melden Sie sich dort mit dem Formular bis 11. September 2011 zu den Tutorien und der Mitgliederversammlung an.

Falls Sie ein Tutorium oder einen Vortrag anbieten wollen, werden Sie gebeten, dies mit dem entsprechenden Formular auf der Internetseite oder per E-Mail

an mv45@dante.de möglichst bald anzumelden. Fügen Sie hierzu bitte eine Kurzzusammenfassung (Abstract) von maximal einer DIN A4-Seite als Text- oder \LaTeX -Datei bei.

Die Adresse für schriftliche Anmeldungen, Fragen, Wünsche und Anregungen rund um die Tagung lautet:

DANTE e.V.

Stichwort: Mitgliederversammlung von DANTE e.V.

Postfach 10 18 40

69008 Heidelberg

E-Mail (bevorzugt): mv45@dante.de

Wir freuen uns auf viele neue und alte Gesichter in Garmisch-Partenkirchen, das mit einem reizvollen touristischen Beiprogramm lockt.

Mit freundlichen Grüßen,

Volker RW Schaa (Vorsitzender DANTE e.V.)

Andreas Hirsch (Werdenfels-Gymnasium)



Foto: Andreas Hirsch

Bretter, die die Welt bedeuten

Variable Argumente in L^AT_EX nutzen

Dominik Wagenführ

Dadurch, dass man in L^AT_EX eigene Befehle definieren kann, lassen sich wiederkehrende Aufgaben bzw. Formatierungen leicht umsetzen, ohne unnötig L^AT_EX-Code mehrfach schreiben zu müssen. Ein zweiter Vorteil bei einem eigenen Befehl besteht darin, dass man bei einer gewünschten Änderung nur eine Textstelle bearbeiten muss und nicht mehrere separate Stellen im Dokument. Der Artikel soll zeigen, wie man mithilfe des Paketes `xkeyval` [2] optionale Argumente bei selbstdefinierten L^AT_EX-Befehlen nutzen kann, ohne den Überblick zu verlieren.

Das Paket `xkeyval` ist dabei eine Erweiterung des Paketes `keyval` [1], mit welchem man einen Großteil der Beispiele des Artikels auch nachvollziehen kann. Daneben gibt es noch `pgfkeys` und `kvoptions/kvsetkeys`, die mit einer Schlüsselbehandlung bei Optionen umgehen können.

Eigene Befehle definiert man durch das Kommando `\newcommand`. Das folgende Minimalbeispiel wird im Laufe des Artikels um neue Befehlsdefinitionen erweitert:

```
\documentclass[parskip=half-]{scrartcl}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
% Blindtext zum Testen
\newcommand*\lorem{Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed
diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed
diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum.}
\newcommand\framedtextA[1]{\fbox{#1}}
\begin{document}
\framedtextA{\lorem}
\end{document}
```

Hier wird der Befehl `\framedtextA` definiert, der genau ein Argument erwartet ([1]) und dieses dann in einer simplen Box umrahmt darstellt. Mittels `#1` kann man auf das Argument zugreifen. In der aktuellen Definition hat dies natürlich den Nachteil, dass `\fbox` die Zeilen nicht automatisch umbricht, sodass man daher den Inhalt lieber in eine Minipage setzt:

```
\usepackage{calc}
\newcommand\framedtextA[1]{%
  \fbox{%
    \begin{minipage}{\linewidth-2\fboxsep-2\fboxrule}%
      #1%
    \end{minipage}}}
```

Damit die Box um die Minipage genau so breit ist, wie die aktuelle Zeile, wird von der aktuellen Zeilenlänge `\linewidth` die Rahmenstärke `\fboxrule` (auf beiden Seiten) sowie der Rahmenabstand `\fboxsep` (ebenfalls auf beiden Seiten) abgezogen. Damit die symbolische Rechnung möglich ist, benötigt man das \LaTeX -Paket `calc`. Als nächstes soll der Befehl erweitert werden, bis er so kompliziert bei der Benutzung wird, dass man sich eine Alternative wünscht.

Rahmenstärke und -abstand angeben

Als erste Erweiterung soll man die Rahmenstärke und den Abstand zum innenliegenden Text angeben können:

```
\newcommand\framedtextB[3]{%
  \begingroup%
  \setlength\fboxrule{#1}%
  \setlength\fboxsep{#2}%
  \fbox{%
    \begin{minipage}{\linewidth-2\fboxsep-2\fboxrule}%
      #3%
    \end{minipage}}%
  \endgroup}
```

Die Benutzung im Dokument wäre dann:

```
\framedtextB{3pt}{10pt}{\lorem}
```

was den gleichen Text wie zuvor in einer Box darstellt, wobei der Rahmen aber eine Breite von 1 Punkt (1pt) hat und der Abstand zum Text 10 Punkte (10pt) beträgt. Für die Befehle wird `\begingroup ... \endgroup` genutzt, damit die Veränderung von `\fboxrule` und `\fboxsep` am Ende eines Befehls automatisch aufgehoben wird.

Textfarbe ändern

Etwas bunter darf es ruhig zugehen, daher soll der Text farbig dargestellt werden. Dafür benötigt man das \LaTeX -Paket `xcolor`:

```
\usepackage{xcolor}
\newcommand\framedtextC[4]{%
  \begingroup%
  \setlength\fboxrule{#1}%
```

```
\setlength\fboxsep{#2}%
\fbox{%
  \begin{minipage}{\linewidth-2\fboxsep-2\fboxrule}%
  \textcolor{#3}{#4}%
  \end{minipage}}%
\endgroup}
```

Die Benutzung ist:

```
\framedtextC{3pt}{10pt}{red}{\lorem}
```

Das erstellt den Text in der Box in roter Farbe. Das Paket `xcolor` wird dem Paket `color` vorgezogen, da es eine weitaus bessere Farbunterstützung bietet und einen kleinen Fehler in `\fcolorbox` korrigiert.

Rahmen- und Hintergrundfarbe ändern

Da der Befehl `\framedtextC` immer noch recht überschaubar ist, soll auch die Rahmenfarbe und die Hintergrundfarbe verändert werden können:

```
\newcommand\framedtextD[6]{%
  \begingroup%
  \setlength\fboxrule{#1}%
  \setlength\fboxsep{#2}%
  \fcolorbox{#4}{#5}{%
    \begin{minipage}{\linewidth-2\fboxsep-2\fboxrule}%
    \textcolor{#3}{#6}%
    \end{minipage}}%
  \endgroup}
```

Die Benutzung ist nun:

```
\framedtextD{3pt}{10pt}{white}{green}{black}{\lorem}
```

was den Text weiß auf schwarzem Hintergrund darstellt und dies alles grün umrahmt.

Textausrichtung festlegen

Nun soll man von außen auch auf rechtsbündigen, linksbündigen oder zentrierten Text umschalten können. Der Einfachheit halber wird dabei mit einem optionalen siebten Argument gearbeitet, welches man in eckigen Klammern nach der Anzahl der Argumente angibt. Für die Entscheidung, welche Ausrichtung gewählt wird, wird das Paket `xifthen` benutzt:

```
\usepackage{xifthen}
\newcommand\framedtextE[7][b]{%
  \begingroup%
  \setlength\fboxrule{#2}%
  \setlength\fboxsep{#3}%
```

```

\colorbox{#5}{#6}{%
\begin{minipage}{\linewidth-2\fbboxsep-2\fbboxrule}%
\ifthenelse{\equal{#1}{r}}{%
\flushright}{%
\ifthenelse{\equal{#1}{l}}{%
\flushleft}{%
\ifthenelse{\equal{#1}{c}}{%
\centering}{%
\ifthenelse{\equal{#1}{b}}{\error{1}}}}}%
\textcolor{#4}{#7}%
\end{minipage}}%
\endgroup}

```

Jetzt wächst der Aufruf zu folgendem Konstrukt an, wenn der Text rechtsbündig ausgegeben werden soll:

```
\framedtextE[r]{3pt}{10pt}{white}{green}{black}{\lorem}
```

Dies kann man jetzt noch eine Weile fortführen, spätestens beim zehnten Argument stößt man aber auf eine natürliche Begrenzung von L^AT_EX, denn man wird dezent darauf hingewiesen, dass man bereits neun Parameter hat:

```
! You already have nine parameters
```

Bei derartig vielen Parametern wäre es besser, wenn die Benutzung etwas intuitiver vonstatten ginge. So wäre es nicht schlecht, wenn man klare Begriffe mit einer Farbe assoziieren könnte, oder wenn die Reihenfolge der Angaben beliebig wäre. Zudem ist es nicht gerade sinnvoll, immer alle Angaben vornehmen zu müssen, selbst wenn man auf Standardwerte zurückgreift. Für neue Umgebungen mittels `\newenvironment` gelten im Übrigen die gleichen Aussagen.

Argumente mithilfe von `xkeyval`

Mit dem L^AT_EX-Paket `xkeyval` [2] kann man alle Probleme umgehen, die sich gestellt haben. Der folgende Abschnitt soll anhand von Beispielen die Benutzung des Paketes aufzeigen, sodass auch ein L^AT_EX-Anfänger relativ schnell zu einem Ergebnis kommt. Für eine ausführliche Beschreibung aller Optionen und Möglichkeiten sollte man sich die Dokumentation auf der Webseite durchlesen.

Bei der Benutzung von `xkeyval` werden die optionalen Argumente nach sogenannten *Schlüssel-Wert-Paaren* durchsucht. Zu jedem Schlüssel wird dann der zugehörige Wert ausgelesen und der L^AT_EX-Schreiber kann den Wert entsprechend weiterverarbeiten. Manchmal wird dieser direkt an ein weiteres L^AT_EX-Kommando weitergegeben, manchmal aber auch erst zwischengespeichert, um später darauf zugreifen zu können. Die Angabe eines Schlüssels und des zugehörigen Wertes geschieht dabei in der Regel über die Angabe *key=value*.

Farbangaben der Box ersetzen

Als Erstes sollen die drei Farbangaben für die Text-, Rahmen- und Hintergrundfarbe so geändert werden, dass man diese als optionales Argument angeben kann. Als Basis für die Änderung wird die Version `\framedtextD` von oben benutzt. Die Möglichkeit zur Ausrichtung des Textes wird dann später wieder eingefügt.

Zuerst sollte man sich für die drei Farben drei neue Befehle definieren, die den Farbnamen enthalten und später überschrieben werden:

```
\newcommand*\TextColor{black}
\newcommand*\BackgroundColor{white}
\newcommand*\BorderColor{black}
```

Danach kann man auch schon mit der Definition der neuen Schlüssel beginnen:

```
\makeatletter
\define@key{TextBox}{textcolor}{\renewcommand*\TextColor{#1}}
\define@key{TextBox}{background}{\renewcommand*\BackgroundColor{#1}}
\define@key{TextBox}{bordercolor}{\renewcommand*\BorderColor{#1}}
\makeatother
```

Dies war es auch schon. Mittels `\define@key` definiert man also einen neuen Schlüssel. Das erste Argument (`TextBox`) gibt dabei die Familie an, die später für die Identifikation benutzt wird. Das zweite Argument steht für die Schlüsselbezeichnung. Als drittes folgt der \LaTeX -Code, in dem man den Schlüssel verarbeitet, auf den man mittels des Argumentes `#1` zugreifen will. Im obigen Fall werden die drei Werte also einfach nur in den vordefinierten Kommandos zwischengespeichert.

Der neue Befehl sieht dann wie folgt aus:

```
\newcommand\framedtextF[4][{}]{%
  \begingroup%
  \setkeys{TextBox}{#1}%
  \setlength\fbxrule{#2}%
  \setlength\fbxsep{#3}%
  \fcolorbox{\BorderColor}{\BackgroundColor}{%
  \begin{minipage}{\linewidth-2\fbxsep-2\fbxrule}%
    \textcolor{\TextColor}{#4}%
  \end{minipage}}%
  \endgroup}
```

Wichtig ist die Zeile

```
\setkeys{TextBox}{#1}%
```

denn über diese gibt man alle optionalen Argumente, die im ersten Argument (`#1`) stehen, an `xkeyval` zur Interpretation. Dieses sucht dann darin nach Schlüsseln und Werten und führt den zugehörigen Code aus. Hierbei ist es wichtig, die

richtige Familie anzugeben, da sonst die Schlüssel nicht erkannt werden. Nach dem Setzen der Schlüssel kann man über die Kommandos `\BorderColor`, `\BackgroundColor` und `\TextColor` auf die Farbdefinitionen zugreifen. Eine Benutzung sieht dann wie folgt aus:

```
\framedtextF[textcolor=white,bordercolor=green,
              background=black]{3pt}{10pt}{\lorem}
```

Dies ist zwar länger als zuvor, hat aber den Vorteil, dass aufgrund der Assoziation mit einem Schlüsselwort wie `textcolor` sofort klar ist, was das Argument beeinflusst. Daneben ist auch die Reihenfolge der Angaben nicht mehr relevant:

```
\framedtextF[background=black,textcolor=white,
              bordercolor=green]{3pt}{10pt}{\lorem}
```

erzeugt die gleiche Box. Da es sich um optionale Werte handelt, kann man natürlich auch nur einen Teil angeben, beispielsweise

```
\framedtextF[textcolor=black]{3pt}{10pt}{\lorem}
```

Fehlende Schlüssel vorbelegen

Will man vorbelegte Standardwerte nutzen, wenn die Angabe einer Option komplett fehlt, fügt man folgende Zeile (am besten unter der Definition der Schlüssel) ein:

```
\presetkeys{TextBox}{bordercolor=black,textcolor=black,background=white}{}%
```

Im folgenden Beispiel

```
\framedtextF[textcolor=red]{3pt}{10pt}{\lorem}
```

werden für die weggelassenen `bordercolor`- und `background`-Angaben die mit `\presetkeys` eingestellten Werte verwendet, so dass roter Text auf weißem Hintergrund mit schwarzem Rahmen steht.

```
\framedtextF[textcolor=red]{3pt}{10pt}{\lorem}
```

Der Befehl `\presetkeys` definiert wie gesagt die Standardwerte, wenn eine Option nicht angegeben ist. Das erste Argument gibt wieder die Familie an. Das zweite und dritte Argument sind ähnlich, denn das zweite setzt die Standardwerte *bevor* die benutzerdefinierten Werte mit `\setkeys` gesetzt werden, das dritte Argument setzt diese erst *danach*. Die Standardwerte werden dabei aber nur gesetzt, falls der Wert nicht als Option vom Benutzer angegeben wurde.

In obigem Beispiel hat die Unterscheidung, ob man die Standardwerte zuerst oder zuletzt setzt, keine Bedeutung. Dies ist aber beispielsweise dann wichtig, wenn man in einer Schlüsseldefinition auf einen anderen Schlüsselwert zugreifen will. Näheres dazu erfährt man in der Dokumentation.

Die Vorbelegung bleibt aktiv, bis sie mit `\presetkeys` erneut überschrieben wird. Das heißt, man kann in jedem Befehl vor `\setkeys` die Vorbelegung anders setzen, wenn das gewünscht ist.

Natürlich muss man `\presetkeys` nicht einsetzen, sondern kann auch selbst für eine Vorbelegung sorgen, indem man vor dem `\setkeys`-Befehl alle Werte selbst mit `\renewcommand` initialisiert. Alternativ kann man auch ein zweites `\setkeys` aufrufen, um Werte einmalig zu setzen.

Boxgrößen ändern

Analog zu den Farben soll natürlich entsprechend mit den Größenangaben zu Rahmenstärke und Rahmenabstand verfahren werden. Hier kann man zuerst mit der Zwischenspeicherung der Längen wie oben beginnen:

```
\newlength{\BorderWidth}
\newlength{\BorderSeparation}

\makeatletter
\define@key{TextBox}{border}{\setlength\BorderWidth{#1}}
\define@key{TextBox}{bordersep}{\setlength\BorderSeparation{#1}}
\makeatother
```

Die Vorbelegung sollte man auch nicht vergessen:

```
\presetkeys{TextBox}{bordercolor=black,textcolor=black,background=white,
border=0.8pt,bordersep=3pt}{}
```

Und natürlich benötigt man noch ein neues Kommando:

```
\newcommand\framedtextG[2][]{%
\begingroup%
\setkeys{TextBox}{#1}%
\setlength\fbxrule{\BorderWidth}%
\setlength\fbxsep{\BorderSeparation}%
\colorbox{\BorderColor}{\BackgroundColor}{%
\begin{minipage}{\linewidth-2\fbxsep-2\fbxrule}%
\textcolor{\TextColor}{#2}%
\end{minipage}}%
\endgroup}
```

Die Benutzung im Standardfall sieht nun ganz einfach aus:

```
\framedtextG{\lorem}
```

Oder mit einem etwas dickeren und grünen Rahmen wie folgt:

```
\framedtextG[bordercolor=green,border=3pt]{\lorem}
```

Eigentlich ist diese Lösung aber umständlich, da man die Längen für `\fbxrule` und `\fbxsep` auch gleich ohne Umwege setzen kann. Dafür entfernt man die

beiden `\newlength`-Zeilen wieder und ersetzt die Definition der Schlüssel sowie den Befehl `\framedtextG` durch:

```
\makeatletter
\define@key{TextBox}{border}{\setlength\fbxrule{#1}}
\define@key{TextBox}{bordersep}{\setlength\fbxsep{#1}}
\makeatother
\newcommand\framedtextG[2][]{%
  \begingroup%
  \setkeys{TextBox}{#1}%
  \fcolorbox{\BorderColor}{\BackgroundColor}{%
  \begin{minipage}{\linewidth-2\fbxsep-2\fbxrule}%
    \textcolor{\TextColor}{#2}%
  \end{minipage}}%
  \endgroup}
```

So wirkt das Ganze schon etwas übersichtlicher.

Fehlende Schlüsselwerte vorbelegen

Neben der Vorbelegung mit Standardwerten, wenn der Schlüssel fehlt, gibt es auch die Besonderheit, einen Schlüssel mit einem Wert vorzubelegen, wenn der Schlüssel angegeben wird, aber kein Wert.

Als sinnvolles Beispiel soll der Rahmen der Box standardmäßig gar nicht angezeigt werden. Nur wenn man mindestens `border` oder `border=WERT` als Option schreibt, soll ein Rahmen angezeigt werden. Dafür ändert man die Schlüsseldefinition wie folgt:

```
\define@key{TextBox}{border}[0.8pt]{\setlength{\fbxsep}{#1}}
```

und die Vorbelegung in

```
\presetkeys{TextBox}{bordercolor=black,textcolor=black,background=white,
border=0pt,bordersep=3pt}{}
```

Das bedeutet also, dass, wenn die Option `border=WERT` bei der Angabe von `\framedtextG` fehlt, kein Rahmen angezeigt wird. Schreibt man dagegen nur `border`, wird der in eckigen Klammern angegebene vordefinierte Wert bei `\define@key` benutzt.

Als kleine Übung kann man den Abstand zum Rahmen ebenfalls per Standard auf `0pt` setzen, bei der Angabe von `bordersep` aber den Wert `3pt` nehmen. Fortgeschrittene L^AT_EX-Nutzer und `xkeyval`-Kenner können versuchen, den Wert bei `bordersep` in Abhängigkeit von `border` zu setzen, denn nur wenn der Rahmen überhaupt sichtbar ist, ist auch ein Rahmenabstand sinnvoll.

Textausrichtung einstellen

Was jetzt noch fehlt, ist die Einstellung der Textausrichtung. Es wäre zwar möglich, diese nach wie vor über simple Buchstaben wie `r`, `l`, `c` oder `b` zu definieren, nur würde man dann für jede dieser Optionen einen eigenen Schlüssel benötigen. Und die Frage ist, was passiert, wenn jemand `r`, `l` als Option angibt. Aus diesem Grund ist es besser, einen Schlüssel `align` zu definieren, der nur bestimmte Werte akzeptiert. Hierfür gibt es den sogenannten `choicekey`:

```
\makeatletter
\define@choicekey{TextAlignment}{align}[\val\al]{right,left,center,block}{%
  \ifcase\al\relax \raggedleft%
    \or\raggedright%
    \or\centering%
    \or% Blocksatz
  \fi}
\makeatother
\presetkeys{TextAlignment}{align=block}{}
```

Alternativ könnte man auch die Verwendung von `align` so ändern, dass der Benutzer immer den echten Befehl angeben muss, also beispielsweise `align=\raggedleft` für rechtsbündigen Text. Dies erfordert aber, dass der Benutzer die genauen L^AT_EX-Befehle kennen muss, und würde außerdem dem Beispiel hier im Artikel die Grundlage entziehen.

Die Argumente von `\define@choicekey` sind als Erstes wieder die Familie, danach das Schlüsselwort und als drittes und optionales Argument die Zuweisung des Wertes (in unserem Beispiel kann man über `\al` darauf zugreifen). Danach folgt die Liste der erlaubten zuweisbaren Werte und zum Schluss der L^AT_EX-Code, der mittels `\ifcase` zu den erlaubten Werten den richtigen Code ausführt. Dabei muss die Reihenfolge der erlaubten Werte mit den auszuführenden Befehlen übereinstimmen. Man sollte bei dem Beispiel aber aufpassen, weil eine neue Familie `TextAlignment` benutzt wird. Den Grund dafür zeigt die Definition des neuen Befehls:

```
\newcommand\framedtextH[2][]{%
  \begingroup%
  \setkeys*{TextBox}{#1}%
  \fcolorbox{\BorderColor}{\BackgroundColor}{%
    \begin{minipage}{\linewidth-2\fbboxsep-2\fbboxrule}%
      \setrmkeys{TextAlignment}%
      \textcolor{\TextColor}{#2}%
    \end{minipage}}%
  \endgroup}
```

Die Schlüssel für die Ausrichtung (englisch »alignment«) dürfen erst innerhalb der `Minipage`-Umgebung gesetzt werden, da die daraus entstehenden Befehle

wie `\raggedright` oder `\centering` außerhalb der Umgebung nicht die richtige Wirkung zeigen würden.

Die Familie `TextBox` kennt aber keinen Schlüssel `align`, daher muss dieser ignoriert werden. Man könnte dies entweder durch die Zeile

```
\setkeys{TextBox}[align]{#1}%
```

erreichen, bei dem alle unbekannt Schlüsselworte in den eckigen Klammern ignoriert werden.

Besser ist aber die Lösung über `\setkeys*` oben. Alle in der Familie unbekannt Werte werden in einer eigenen Liste gespeichert, die man dann später mittels `\setrmkeys` der richtigen Familie zuweisen kann (`»rm«` steht für `»remaining«`).

Von der unsauberen Lösung

```
\setkeys{TextBox,TextAlignment}{#1}%
```

anstelle des `\setrmkeys` sollte man absehen, da man im Allgemeinen nicht sicherstellen kann, dass die Schlüsselzuweisung von `TextBox` keine negativen Nebeneffekte auf den L^AT_EX-Code hat.

Texte mit mehreren Absätzen

Obige Definition von `\framedtextH` hat den Nachteil, dass es keine Texte mit mehreren Absätzen zulässt. Als Verbesserung kann man folgende Definition nutzen:

```
\newsavebox\MyFrameBox
\newcommand\framedtextI[2][]{%
  \begingroup%
  \setkeys*{TextBox}{#1}%
  \sbox{\MyFrameBox}{%
    \begin{minipage}{\linewidth-2\fbboxsep-2\fbboxrule}%
      \setrmkeys{TextAlignment}%
      \color{\TextColor}%
      #2%
    \end{minipage}%
  }%
  \fcolorbox{\BorderColor}{\BackgroundColor}{\usebox{\MyFrameBox}}%
  \endgroup}
```

Zunächst wird der Text im Boxregister `\MyFrameBox` gesetzt und letzteres dann in `\fcolorbox` verwendet. Zusätzlich wurde von `\textcolor` auf `\color` umgestellt. Dadurch kann der Text mehrere Absätze enthalten, was sonst `\fcolorbox` und `\textcolor` verhindert hätten.

```
\framedtextI[border]{\lorem\par\lorem}
```

Weitere `xkeyval`-Schlüssel

Es gibt noch zwei weitere Befehle, die für den einen oder anderen Benutzer wichtig sein könnten. Zum einen

```
\define@cmdkey{FAMILIE}{SCHLUESSEL}[STANDARDWERT]{LATEX-CODE}
```

Dies definiert analog zu oben einen normalen Schlüssel, zusätzlich hat man innerhalb des \LaTeX -Codes (letztes Argument) über das Kommando `\cmdKV@FAMILIE@SCHLUESSEL` Zugriff auf den Inhalt des definierten Schlüssels.

Der zweite Befehl definiert boolesche Schlüssel:

```
\define@boolkey{FAMILIE}{SCHLUESSEL}[STANDARDWERT]{LATEX-CODE}
```

Diese ähneln `choicekey`, denn es sind nur die Schlüsselwerte `true` und `false` erlaubt. Man kann sich zum Beispiel eine neue boolesche Variable definieren

```
\newboolean{Variable}
```

und im \LaTeX -Code Folgendes schreiben:

```
\setboolean{Variable}{#1}
```

Danach kann man in seinem Code entsprechend des Wertes unterschiedliche Befehle ausführen:

```
\ifthenelse{\boolean{Variable}}{%
  % Code, falls Variable true ist
}{%
  % Code, falls Variable false ist
}
```

Abschließende Bemerkung

Wie oben angekündigt, geben die im Artikel vorgestellten Beispiele nur einen kleinen Überblick über das, was man mit `xkeyval` machen kann. Dennoch kann man auch als \LaTeX -Anfänger sehr schnell gute Resultate erzielen, wenn man das recht einfache Prinzip verstanden hat. Und selbst, wenn man nur Copy & Paste nutzt, kommt man womöglich zum Ziel, auch ohne die Details zu verstehen.

Literatur

- [1] David Carlisle: *The keyval package*; CTAN:macros/latex/required/graphics/keyval.dtx; 1999.
- [2] Hendry Adriaens: *The xkeyval package*; CTAN:macros/latex/contrib/xkeyval/; 2008.

Viele Ziele – Multi Target Publishing

Axel Kielhorn

Ein Weg führt zu einem Ziel

Bisher war das Ziel meiner Veröffentlichungen immer aus Papier und in einem bestimmten Format: DIN A4, DIN A5 oder auch mal 9 cm × 12 cm. Auf dem Weg dahin entstand allzeit eine PDF-Datei, daher lag es nahe, diese auch am Bildschirm zu lesen, was zumindest bei DIN A5 bequem möglich ist. Doch dann kamen die Mobilgeräte. Einige davon können auch PDFs anzeigen und mit etwas Aufwand kann man den Text so formatieren, dass er auf *einem* Mobilgerät mit wenig Scrollen lesbar ist.

Günstiger wäre natürlich ein Format, bei dem der Leser die Textgröße bestimmen kann und das Gerät den Text passend zur Anzeigengröße umbricht. Ein solches Format ist beispielsweise EPUB. Im Prinzip ist das nichts anderes als ein ZIP-Archiv mit einer definierten Struktur und einigen XML-Dateien, die den eigentlichen Text enthalten. Das Aussehen kann durch eine css-Datei gesteuert werden.

Eine Umleitung

Glücklicherweise gibt es ein Programm, das \LaTeX lesen und EPUB schreiben kann: `pandoc`¹. [2] Wenn die \LaTeX -Datei nicht zu kompliziert ist, kann `pandoc` sie verstehen und konvertieren. Aber was ist zu kompliziert? Am einfachsten konvertiert man ein Dokument von » \LaTeX « nach \LaTeX « und sieht, was übrig bleibt:

```
pandoc -r latex -t latex --template=./latex-de.template -o quelle-pd.tex quelle.
tex
```

Dieser Befehl benutzt eine an die deutsche Sprache angepasste Version der Standardvorlage `latex.template`, um eine neue \LaTeX -Datei zu erstellen.

Die Steine auf dem Weg

`pandoc` arbeitet mit UTF-8-kodierten Dateien. Latin-1-kodierte Texte lassen sich leicht konvertieren, Umlautumschreibungen gemäß `german.sty` gehen jedoch verloren. Auch \TeX -Akzente `\^o` oder Einbuchstabenbefehle `\o` führen manchmal zu Problemen. Das lässt sich aber mit ein paar Zeilen `sed` beheben (siehe Abschnitt Reisevorbereitung). Einfache Auszeichnungsbefehle für **fett** und *kursiv* werden unterstützt, doch schon bei geschachteltem `\emph` ist Schluss.

¹ `pandoc` ist in der Version 1.8.1.3 unter GPL lizenziert.

Ein neuer Anfang

Steht die \LaTeX -Datei wirklich am Anfang? Oder sollten wir nicht lieber, wie von Martin Schröder auf der Dante-Tagung in Bremen angedeutet, \LaTeX als *ein* Backend verstehen und die \LaTeX -Datei somit nur als ein Zwischenprodukt?

Eine ungewöhnliche Richtung: Markdown statt markup

Markdown ist eine von John Gruber [1] entwickelte Auszeichnungssprache, die jedoch so aussieht, als ob sie keine Auszeichnungen enthält:

A Markdown-formatted document should be publishable as-is, as plain text, without looking like it's been marked up with tags or formatting instructions.

Der Anfang dieses DTK-Artikels sieht in Markdown so aus:

```
# Ein Weg führt zu einem Ziel
```

```
Bisher war das Ziel meiner Veröffentlichungen immer aus Papier und in einem bestimmten Format: DIN A4, DIN A5 oder auch mal 9 cm  $\times$  12 cm. Auf dem Weg dahin entstand immer eine PDF-Datei, daher lag es nahe, diese am Bildschirm zu lesen, was zumindest bei DIN A5 bequem möglich ist.
```

```
Doch dann kamen die Mobilgeräte. Einige davon können auch PDFs anzeigen und mit etwas Aufwand kann man den Text so formatieren, das er auf *einem* Mobilgerät mit wenig Scrollen lesbar ist.
```

Dieser Text wurde mit dem Befehl

```
pandoc -r latex -t markdown -o Ziele.md Ziele.tex
```

direkt aus der \LaTeX -Datei erzeugt.

Markdown ist eine sehr eingeschränkte Sprache, die Manpage, die den Sprachumfang beschreibt, ist nur 16 Seiten lang (im Vergleich dazu ist \LaTeX dreimal so lang) und kratzt gerade mal an der Oberfläche von \LaTeX . Bei der Konvertierung von \LaTeX zu Markdown sollte man außerdem beachten, dass pandoc kein \TeX versteht. Es nutzt reguläre Ausdrücke, um \LaTeX zu interpretieren. Das hat zur Folge, dass bei einigen Befehlen zusätzliche Leerzeilen erforderlich sind, damit pandoc die Befehle von normalem Text unterscheiden kann.

Auf neuem Weg zum alten Ziel: PDFs mit Markdown über \LaTeX generieren

```
pandoc -r markdown -t latex --template=./latex-de.template -o quelle.tex quelle.md
```

Die `latex.template` Dateien

Die mitgelieferte Datei `latex.template` ist eine Minimalversion. Für deutsche Texte empfiehlt sich die erweiterte Datei `latex-de.template`, die im Begleitmaterial zu finden ist. [7] Die Änderungen zur Originalversion sind durch ein `-ak` gekennzeichnet.

Abkürzung

Der schnellste Weg, aus einer Markdown-Datei ein PDF-Dokument zu erstellen, ist:

```
markdown2pdf --template=./latex-de.template quelle.md
```

Im Hintergrund wird hier natürlich auch eine \LaTeX -Datei erstellt und `pdf \LaTeX` aufgerufen, um die PDF-Datei zu erstellen. Mit den Schaltern `--xetex` bzw. `--luatex` kann man stattdessen auf $X_{\text{e}}\LaTeX$ oder $\text{Lua}\LaTeX$ umschalten. In der Datei `latex.template` wird mit Hilfe von `iftex` ermittelt, welches $\text{T}_{\text{E}}\text{X}$ verwendet wird.

Nacharbeit

Die automatisch erstellte \LaTeX -Datei ist fast so gut, wie eine von einem Anfänger manuell erstellte. Natürlich wird man auch hier mit Trennhilfen und Ähnlichem die *overflow* und *underfull hboxes* beseitigen müssen.

Der Weg zu einem neuen Ziel: EPUB

Der ursprüngliche Wunsch war eine EPUB-Datei zu erstellen. Dies geschieht mit dem Befehl:

```
pandoc -r markdown -t epub --epub-cover-image=cover-image.gif -s -o Ziele.epub
Ziele.md
```

Der Text wird anhand der Gliederungsbefehle in verschiedene Dateien aufgeteilt, das erleichtert die Nachbehandlung, beispielsweise mit `Sigil`. [6] Seit der Version 1.8.1.2 kann ein Titelbild mit der Option `--epub-cover-image` eingebunden werden.

Der Weg zu OpenOffice

»Kann ich das als Word-Datei haben?« Wer kennt diese Frage nicht? Treffen wir uns auf halben Weg bei einer OpenOffice-Datei.²

²Mit `Writer2LaTeX` kann man eine OpenOffice Datei wieder in \LaTeX zurückverwandeln, Umwandlungsverluste inklusive.

```
pandoc -r markdown -t odt --reference-odt=./reference-de.odt -s -o quelle.odt
quelle.md
```

Die Datei `reference-de.odt` dient hier als Basis für die Absatzvorlagen und das Dokumentlayout. Dies kann eine beliebige OpenOffice-Datei sein, am einfachsten ist es jedoch, eine von pandoc erstellte Datei an die eigenen Layoutwünsche anzupassen. So ist sichergestellt, dass die internen Bezeichnungen der Absatzformate mit den von pandoc vergebenen übereinstimmen.

Die Datei `reference-de.odt` basiert auf einer von pandoc erstellten Datei, bei der auf die deutsche Sprache umgestellt wurde. Bis pandoc 1.8.1.2 gab es ein Problem beim Einbinden von Bildern. OpenOffice meldete eine beschädigte Datei und bot an, diese zu reparieren. Dieser Fehler wurde mit der Version 1.8.1.3 behoben. Eingebundene Bilder werden auf eine Einheitsgröße verkleinert und müssen jeweils einzeln auf Originalgröße expandiert werden.

Reisevorbereitung

Mit einem kleinen `sed`-Skript lässt sich diese `dtk`-basierte \LaTeX -Quelle in eine neutrale \LaTeX -Datei umwandeln, die von pandoc weitgehend ohne Verluste gelesen werden kann. Lediglich die `bibliography`-Umgebung geht verloren.

```
s/\LaTeX/LaTeX/g
s/\TeX/TeX/g
s/\ConTeXt/ConTeXt/g
s/\Program{([a-zA-Z]*)}/\1/g
s/\Acronym{([a-zA-Z]*)}/\1/g
s/\Package/\texttt/g
s/[style=DTK\lstNoNumber\]//
s/\File/\texttt/g
s/\Macro{\texttt{\textbackslash }/g
s/\begin{group//
s/\end{group//
```

Mit dem Aufruf

```
sed -f dtk2mdtex.sed Ziele.tex >Ziele-clean.tex
```

werden die \LaTeX -Befehle, die pandoc nicht kennt, aus der Quelldatei entfernt. Das Ergebnis kann dann mit

```
pandoc -r latex -t markdown -s -o Ziele-clean.md Ziele-clean.tex
```

nach Markdown konvertiert werden.

Wegbeschreibung

Gliederungsbefehle

Markdown unterstützt bis zu sechs Gliederungsebenen. Die jeweilige Gliederungsstufe wird durch die Anzahl der # Zeichen vorgegeben. Vor jeder Überschrift ist eine Leerzeile erforderlich.

```
# Oberste Ebene

## Zweite Ebene

### Dritte Ebene

#### *Wichtige Information* in der vierten Ebene versteckt
```

Eine alternative Form der Gliederungsbefehle unterstützt nur zwei Ebenen:

```
Erste Ebene
=====

Zweite und letzte Ebene
-----
```

Zitatumgebung

Zitiert wird wie in (alten) E-Mail-Programmen mit einem >

```
> Hier zitiere ich eine alte E-Mail.
>
> > Und hier wird in der alten E-Mail eine noch ältere zitiert.
>
> Normalerweise erhält jede Zeile ein >-Zeichen. Einige Editoren
> fügen dann in der Folgezeile das > gleich mit ein.
>
> Sollte ein Editor nicht über diese Funktion verfügen,
reicht es, nur die erste Zeile eines Absatzes zu markieren.
```

Eine besondere Form von Zitaten sind Zitate aus Programmen, die standardmäßig in einer nichtproportionalen Schrift gesetzt werden. Programmzitate werden durch vier Leerzeichen oder einen Tabulatorschritt eingegeben:

```
    \documentclass[11,ngerman]{dtk}
    \usepackage[utf8]{inputenc}
```

Auch vor Zitaten ist eine Leerzeile erforderlich. Will man bei längeren Listings nicht jede Zeile einrücken, so gibt es in pandoc eine alternative Form, diese ist jedoch nicht Standard-Markdown. Durch eine Zeile mit mindestens drei ~-Zeichen wird das Programmlisting eingeleitet und durch eine Zeile mit mindestens genauso vielen ~-Zeichen wieder beendet. Eine Zeile mit weniger ~-Zeichen ist Bestandteil des Listings.

```

~~~~~
Dies ist ein Programm Listing
~~~~~
Header
~~~~~
Body
~~~~~

```

Listen

Aus \LaTeX kennen wir verschiedene Listenformen:

Die itemize-Liste

Eine Liste wird durch ein Aufzählungszeichen (*, + oder -) eingeleitet.

```

* eins
* zwei
* drei
  - drei a
  - drei b
* vier
  Wie immer verstecken wir die wichtigen Informationen ganz unten,
  in der Hoffnung, dass niemand sie liest.

```

Besteht ein Listeneintrag aus mehreren Absätzen, so sind die Folgeabsätze mit vier Leerzeichen oder einem Tabulatorschritt einzurücken.

```

* eins
* zwei
* drei
  - drei a
  - drei b
* vier

    Wie immer verstecken wir die wichtigen Informationen ganz unten,
    in der Hoffnung, dass niemand sie liest.

    Um ganz sicher zu gehen, erwähnen wir erst im letzten Absatz die
    Vier-Leerzeichen-Regel.

```

Die enumerate-Liste

Das erste Aufzählungszeichen gibt hier das Format vor (1., (1) oder i.). Es ist nicht notwendig, dass die weiteren Aufzählungszeichen in der richtigen Reihenfolge erscheinen (auch wenn das seltsam aussieht). Es wird automatisch das enumerate-Paket geladen. Möchte man das verhindern, so kann man stattdessen die Aufzählungspunkte mit #. markieren, dann wird die Standardeinstellung der verwendeten Dokumentklasse benutzt.

1. ein
2. zwei
4. drei
 - a) drei a
 - b) drei b
5. vier

Wie immer verstecken wir die wichtigen Informationen ganz unten, in der Hoffnung, dass niemand sie liest.

Die description-Liste

Endlich kommen wir zu den beliebten `\l2kurz`-Tieren. Das beschriebene Objekt steht allein in einer Zeile, die Beschreibung wird durch einen Doppelpunkt oder eine Tilde eingeleitet. Eine Beschreibung kann mehrere Absätze umfassen, Folgeabsätze werden durch vier Leerzeichen eingeleitet, es können auch mehrere Beschreibungen zu einem Objekt gegeben werden. Semantisch ist das ein Unterschied, beim \LaTeX -Export werden beide Fälle gleich behandelt.

Gelse
: Ein kleines Tier, das östlich des Semmering Touristen verjagt.

Gemse
: Ein großes Tier, das westlich des Semmering von Touristen verjagt wird.

Nun ein langer Absatz zu der Frage, ob es Gensen oder Gämsen heißt.

Gürteltier

~ ein mittelgroßes Tier, das hier nur wegen der Länge seines Namens vorkommt.

~ Gürteltiere sind in Österreich außerhalb von Zoologischen Gärten selten anzutreffen.

Die fortlaufende Liste

Dies ist eine pandoc-Erweiterung.

Normalerweise fängt jede neue Liste wieder bei eins an. Es gibt jedoch eine besondere Form, die fortlaufend über das gesamte Dokument läuft, vergleichbar mit den `caption`-Zählern für die Umgebungen `figure` oder `table`. Auf diese Zähler kann später Bezug genommen werden.

(@Behauptung) Hier behaupte ich etwas.

Die Behauptung (@Behauptung) wird in (@Beweis) bewiesen.

(@Beweis) Und hier der versprochene Beweis.

Diese Liste benutzt nicht den \LaTeX `\label/\ref`-Mechanismus und erfordert daher keinen zweiten \LaTeX -Lauf.

Tabellen

Dies ist eine pandoc-Erweiterung. Seit der Version 1.8.1.2 werden Tabellen mit dem `ctable`-Paket gesetzt. Dadurch hat sich die Qualität der Ausgabe erheblich verbessert.

Es gibt drei Arten von Tabellen. Bei der Eingabe von Tabellen sollte man auf Tabulatoren verzichten und die Spalten mit Leerzeichen ausrichten.

Rechts	Links	Mitte	Standard
12	12	12	12
123	123	123	123
ab	ab	ab	ab

Table: Eine einfache Tabelle

Der Tabellenkopf und die einzelnen Tabellenzeilen müssen in eine Zeile passen. Die Ausrichtung der Spalten hängt von der Unterstreichung der Kopfzeile ab.

- Ist die Unterstreichung rechtsbündig und steht auf der linken Seite hervor, ist die Ausrichtung rechtsbündig.
- Ist die Unterstreichung linksbündig und steht auf der rechten Seite hervor, ist die Ausrichtung linksbündig.
- Steht die Unterstreichung auf beiden Seiten hervor, ist die Ausrichtung zentriert.
- Ist die Unterstreichung genauso lang wie der Text, wird die Standardeinstellung (linksbündig) benutzt.

Eine Tabelle wird durch das Schlüsselwort `Table:` (oder nur `:`) eingeleitet und muss mit einer Leerzeile abgeschlossen werden. Optional darf ein Tabellentitel vergeben werden. Wird ein Titel verwendet, so wird die Tabelle als Gleitobjekt in einer `table`-Umgebung formatiert, ansonsten als Tabelle im Fließtext. Mehrzeilige Tabellen müssen durch jeweils eine Reihe von Bindestrichen eingeschlossen werden. Die Tabellenzeilen werden durch Leerzeilen getrennt.

Zentrierter Kopf	Standard Kopf	Rechtsbündig	Linksbündig
Erste	Zeile	12.0	Beispiel einer mehrzeiligen Zeile
Zweite	Zeile	5.0	Noch eine mehrzeilige Zeile, durch eine Leerzeile abgetrennt

Beim dritten Typ handelt es sich um sogenannte Rastertabellen. Der äußere Rahmen wird natürlich bei \LaTeX -Ausgabe nicht gesetzt. In den Tabellenzellen können beispielsweise enthalten sein:

Frucht	Preis	Vorteile
Banane	\$1.34	- eingebaute Verpackung - leuchtende Farben
Orange	\$2.10	- heilt Skorbut - ist lecker

Titelei

Dies ist eine pandoc-Erweiterung.

Angaben zu dem Titel des Dokuments, den Autoren und dem Datum der Erstellung werden am Anfang der Datei angegeben:

```
% Viele Ziele Veröffentlichung
% Axel Kielhorn
% DTK 2011-3
```

Lange Titel und mehrere Autoren bricht man auf mehrere Zeilen um:

```
% Viele Ziele Veröffentlichung\
  (Multi Target Publishing)
% Axel Kielhorn
  Babel Fisch (Übers.)
% DTK 2011-3
```

Der `\` am Ende der ersten Zeile erzeugt ein `\\` in der \LaTeX -Ausgabe.

Fußnoten

Dies ist eine pandoc-Erweiterung.

Fußnoten bestehen aus zwei Teilen, der Fußnotenmarkierung und dem Fußnotentext.

```
Dies ist eine Fußnotenmarkierung[^1] und dies eine weitere[^fussnote]

[^1]: Hier ist der Fußnotentext

[^fussnote]: Diese Fußnote ist etwas länger.

  Sie enthält einen zweiten Absatz.
```

Hervorhebungen

Kursiver Text wird durch *ein* * oder _ eingeschlossen. Fettgedruckter Text wird durch *zwei* * oder _ eingeschlossen. Sollen nur Wortteile hervorgehoben werden, ist zwingend * zu nutzen, da in Variablenamen häufig _ als Bestandteil des Namens vorkommt.

Dieser Text wurde _mit dem Unterstrich hervorgehoben_ und dieser *mit Sternchen*.

Für __fetten Test__ benutzt man **zwei** Zeichen.

Variablen_namen_ können Unterstriche enthalten, daher benutzt man zum hervor*heben* das Sternchen.

Zum Hochstellen benutzt man das bekannte ^, zum Tiefstellen muss auf die ~ ausgewichen werden, da der Unterstrich schon zum Auszeichnen verwendet wird. Die Befehle schließen das Argument ein.

H-2~0 ist Wasser, 2^10^ ist 1024.
2^2^2^ ist 2^22^.

Die letzte Zeile mag verwundern, allerdings handelt es sich hier nicht um einen Mathematikatz, sondern um Textexponenten und -indizes.

Mathematik

Dies ist eine pandoc-Erweiterung.

Inline-Mathematik wird, wie aus L^AT_EX bekannt, in \$ eingeschlossen. Der Inhalt wird direkt an L^AT_EX weitergeben, daher ist alles erlaubt, was in L^AT_EX möglich ist.

$$2^{2^{2}} \neq 2^{22}$$

Bei anderen Ausgabeformaten hängt es vom jeweiligen Format und den Optionen beim Aufruf von pandoc ab, was als Ausgabe erzeugt wird. Abgesetzte Formeln können als rohes L^AT_EX eingegeben werden.

Rohes L^AT_EX

Dies ist eine pandoc-Erweiterung.

Alles, was zwischen einem \begin/\end-Paar steht, wird direkt an L^AT_EX (oder ConT_EXt) weitergegeben und in allen anderen Formaten ignoriert.

Rohes HTML

Da Markdown ursprünglich zum Erstellen von HTML-Seiten entwickelt wurde, bietet es die Möglichkeit, HTML direkt einzugeben. Bei nicht auf HTML basierten Ausgabeformaten wird es ignoriert.

Links

Bei einem Format, das ursprünglich zum Erstellen von Webseiten gedacht war, sollte man annehmen, dass es mit Hyperlinks umgehen kann. Und so ist es natürlich auch. Alles, was in einer Zeile zwischen spitzen Klammern steht, ist ein Link.

```
<http://johnmacfarlane.net/pandoc/>
```

Ein Link kann natürlich auch im Text auftauchen:

```
Die Dokumentation zu pandoc befindet sich auf der
[pandoc Webseite](http://johnmacfarlane.net/pandoc/).
```

Bilder

Bilder sind eine besondere Form des Links. Wenn vor dem Link ein ! steht, wird dieser als Link auf ein Bild interpretiert.

```
![Ein Blaues Bild](blau.jpg "Blaues Bild")
```

Das Bild wird in einer Figure-Umgebung gesetzt, der Text in den eckigen Klammern wird als Bildunterschrift verwendet. Möchte man das Bild im Fließtext einbinden, darf der Link nicht allein in einer Zeile stehen.

```
Das ![Rotes Quadrat](rot.png "rote Quadrat") im Fließtext.
```

Es gibt keine Möglichkeit, die Bilder zu skalieren, sie müssen in der richtigen Größe und Auflösung vorliegen.

Am Wegesrand

pandoc kann auch ConTeXt-Dateien erstellen. Damit könnte man vorhandene L^AT_EX-Dateien konvertieren, um den Einstieg in ConTeXt zu finden. ConTeXt kann mithilfe des Filter-Moduls sogar direkt Markdown bearbeiten, indem es pandoc als Filter aufruft. Näheres hierzu gibt es im pandoc-Extra-Wiki. [5]

Ausblick

Versteckt man den gesamten Prozess hinter markdown2pdf ist es ohne L^AT_EX-Kenntnisse möglich, PDF-Dateien mit L^AT_EX zu erstellen. »Das ist mir zu kompliziert« sollte somit kein Argument mehr sein. Für aufwändige Arbeiten mit mehreren Fußnotenapparaten und Literaturverzeichnissen ist pandoc sicherlich unterdimensioniert. Einfache Literaturverzeichnisse sind jedoch möglich, hier wird sogar bib_latex unterstützt. Für den nicht-mathematischen Normalanwender bringt pandoc jedoch eine enorme Vereinfachung und reduziert den Lernaufwand erheblich. Die Hauptarbeit steckt in den template-Dateien, diese müssen einmal

von einem L^AT_EX-Experten erstellt werden. Im pandoc-Wiki [5] finden sich einige erweiterte Vorlagen.

Anhang

Im Begleitmaterial, welches auf <http://www.dante.de/DTK/Software.html> zum Herunterladen liegt, befinden sich folgende Dateien:

md-test.md	Die Beispiele aus diesem Artikel
md-test.tex	Konvertiert nach L ^A T _E X
md-test.pdf	Mit pdfL ^A T _E X gesetzt
md-test.epub	Konvertiert nach EPUB
latex-de.template	Eine L ^A T _E X-Vorlage zum Erstellen deutscher Texte
reference-de.odt	Eine OpenOffice-Vorlage zum Erstellen deutscher Texte
dtk2mdtex.sed	Eine sed Datei zum Entfernen der dtk-spezifischen Auszeichnung
pandoc.pdf	Die pandoc-Manpage
pandoc-markdown.pdf	Die Manpage, die die Markdown-Syntax beschreibt
markdown2pdf.pdf	Die markdown2pdf-Manpage

Literatur

- [1] Daring Fireball: *Markdown*; <http://daringfireball.net/projects/markdown/>; 2011
- [2] *Pandoc – About pandoc*; <http://johnmacfarlane.net/pandoc/>; 2011
- [3] *Pandoc User Manuals (1)* – Manpage zu pandoc. (Als PDF im Begleitmaterial enthalten.)
- [4] *Pandoc Markdown (5)* – Manpage zur Markdown-Implementierung von pandoc. (Als PDF im Begleitmaterial enthalten.)
- [5] *Pandoc Extras*; <https://github.com/jgm/pandoc/wiki/Pandoc-Extras>; Juli 2011
- [6] *Sigil – A WYSIWYG ebook editor*; <http://code.google.com/p/sigil/>; 2011
- [7] *Ziele.zip* – Begleitmaterial zu diesem Artikel gibt es unter <http://www.dante.de/DTK/Software.html>

Installation der T_EX Live 2011 auf Ubuntu¹

Enrico Gregorio

Kann man T_EX Live 2011 auf Ubuntu, einer Linux-Distribution, installieren? Ja, und hier wird gezeigt, wie das geht.

Einführung

Ein Hauptnachteil der T_EX-Paketierung in Ubuntu-Systemen liegt darin, dass das Verwaltungsprogramm `tlmgr` fehlt, was ausdrücklich von den Debian-Entwicklern so gewollt ist. Auf der einen Seite schützt das den Benutzer vor möglichen Schäden an dem System; auf der anderen Seite verhindert das aktive Aktualisierungen der T_EX-Distribution, um etwa Bugs zu korrigieren oder neue Funktionen zu erhalten, was fast jeden Tag geschieht. Mehr noch, im Juli 2011 ist das T_EX Live von Ubuntu (oder Debian sollte man sagen) die 2009er Ausgabe mit dem unveränderten Versionsstand vom Oktober 2009.

Wir beschreiben hier eine Methode, wie man T_EX Live 2011 auf einem Ubuntu-Desktop-System installiert; dieses T_EX Live wird sein `tlmgr` für Verwaltung und Updates behalten. Und, was sehr wichtig ist, es wird nichts Gefährliches in System-Verzeichnisse gesteckt. Nur zwei (oder drei) Dateien werden etwa `/etc` in Unterverzeichnisse abgelegt, die bereits für Systemanpassungen vorgesehen sind. Um zum ursprünglichen T_EX des Systems zurückzukehren, kann die gesamte T_EX Live-Installation mit einem einfachen Befehl gelöscht werden:

```
$ rm -fr /usr/local/texlive/2011
```

Optional entfernt man noch die unter `/etc` hinzugefügten Dateien.

Das Verfahren, wie wir es beschreiben werden, kann mit geeigneten Änderungen an andere Distributionen wie Fedora und OpenSUSE angepasst werden. Bei anderen auf Debian basierenden Distributionen sollte man wie bei Ubuntu vorgehen können. Wir wählen Ubuntu, da dieses eine häufig benutzte Distribution unter den Linux-Desktop-Systemen zu sein scheint.

Eine italienische Version dieses Artikels erscheint auch in *ArsT_EXnica*. Ich hoffe, dass dieser Artikel zu einem Installationsprogramm wird: Es sollte für einen Unix-Guru nicht allzu schwer sein, diese Methode in Skripts für die verschiedenen Distributionen umzusetzen.

¹ Übersetzt von Heiko Oberdiek (die englische Version erschien in TUGboat 32:1, 2011).

Eine kurze Einführung zur Kommandozeile

Der Arbeitsablauf erfordert eine gewisse Vertrautheit mit dem Terminal, auch Konsole genannt, Kommandozeilen einzugeben. Wer sich darunter nichts vorstellen kann, sollte hier aufhören zu lesen. Jedoch ist es nicht so schwer, die Befehle genauso abzuschreiben, wie sie hier angegeben sind. Auch gibt es viele Einführungen zur Kommandozeile, beispielsweise

<https://help.ubuntu.com/community/UsingTheTerminal>

Im Folgenden bezeichnet eine Zeile

```
$ ls -l
```

einen Befehl, der in einem Terminal eingegeben wird, gefolgt von der Eingabetaste, um den Befehl auszuführen. Das Symbol \$ stellt die Eingabeaufforderung des Systems dar, auf Ihrem Terminal kann das anders aussehen, beispielsweise etwa so:

```
enrico@ubuntu:~$
```

Normalerweise folgt nach diesen Eingangsbuchstaben ein blinkendes Kästchen, Cursor genannt. Die Befehlszeile, die im Terminal eingegeben wird, steht nach dem Leerzeichen, das dem \$-Zeichen folgt. Andere \$-Zeichen, die nicht am Anfang einer Zeile stehen, gehören zur Befehlszeile und müssen eingetippt werden. Manchmal sind die Befehle zu lang, als dass sie in eine Zeile dieses Artikels passen; dann werden diese mit Hilfe des umgekehrten Schrägstriches folgendermaßen umbrochen:

```
$ command a b c \  
  d e f
```

Dieser Backslash am Ende der Zeile besagt, dass der Befehl auf der nächsten Zeile fortgesetzt wird. Im Terminal schreibt man alles in eine Zeile ohne die umgekehrten Fortsetzungs-Schrägstriche am Zeilenende. Nicht weglassen darf man Leerzeichen vor diesem Backslash.

Antworten des Systems werden hier unterstrichen und ohne das \$-Symbol dargestellt, zum Beispiel sagt

```
bash: tix: command not found
```

aus, dass das System den Befehl tix ausgeführt haben würde, dieser aber nicht gefunden werden konnte. Der Vorspann »bash:« bezeichnet die *Shell*, die die Befehle auszuführen versucht; ignoriere diese Details. Fast immer ist es nicht notwendig, die verschiedenen Teile der Kommandozeile in Gänze einzutippen: Durch Drücken der Tabulatortaste versucht das System selbst eine Vervollständigung, wenn eine eindeutige gefunden werden kann.

Letzter Rat: Wenn Ihre Tastatur keine »~«-Taste hat, erkundigen Sie sich, wie man dieses Zeichen eingibt, beziehungsweise verwenden Sie so bald als möglich eine Tastatur mit diesem Zeichen.

► Bemerkungen für fortgeschrittene Benutzer werden durch ein Dreieck eingeleitet. Der Artikel geht davon aus, dass als Standard-Shell die `bash` verwendet wird. Wer eine andere Shell verwendet, gilt als »fortgeschrittener Benutzer« und sollte fähig sein, die hier angegebenen Befehle an seine eigene Shell anzupassen.

Vorbereitungen

Installieren Sie `Perl-Tk` und `Perl-doc` mit `Synaptic`. Dann öffnen Sie ein Terminalfenster und bereiten ein Arbeitsverzeichnis vor, beispielsweise

```
$ mkdir ~/texlive-install
$ cd ~/texlive-install
```

Der letzte Befehl wechselt in das Verzeichnis, das zuvor angelegt wurde. Nun werden wir kurz die zwei Hauptwege beschreiben, wie man die \TeX Live erhalten kann: Beim ersten muss man während der Installation online mit dem Internet verbunden sein, der zweite Weg kann auch offline beschriftet werden.

Beschaffung der Distribution (online)

Am einfachsten installiert man \TeX Live über das Internet. Das benötigte komprimierte Archiv des Installationsprogrammes wird durch folgende Eingabe auf der Konsole heruntergeladen:

```
$ wget http://mirror.ctan.org/systems/texlive/tlnet/install-tl-unx.tar.gz
```

Nun wird die heruntergeladene Datei ausgepackt, der Befehl lautet

```
$ tar xzf install-tl-unx.tar.gz
```

Er erzeugt ein neues Verzeichnis, in das wir nun wechseln:

```
$ cd install-tl-20110803
```

Der letzte Teil des Namens ist das Datum, wann das Installationsprogramm erstellt wurde, weshalb er unterschiedlich sein kann. Das zuvor erwähnte Vervollständigungs-Feature mit der Tabulatortaste hilft, um den richtigen letzten Namensteil zu erhalten. Gehen Sie zum Abschnitt »Installation der Distribution«.

Beschaffung der Distribution (offline)

Wenn Ihre Verbindung zum Internet nicht ausreicht oder wenn die Installation später auf einem Rechner ohne Internetverbindung erfolgen soll, dann kann man ein ISO-Abbild der Distribution herunterladen, das ist die DVD quasi als Datei. Die Webadresse lautet

<http://mirror.ctan.org/systems/texlive/Images/texlive2011.iso>

Es handelt sich um eine 2,3-GiB-Datei¹, die man auf einem USB-Stick ablegen kann, um sie auf den Computer zu kopieren, auf dem die Installation gewünscht wird. Oder man kann mit dem ISO-Abbild eine DVD brennen und diese dann auf dem Computer verwenden. Es gibt auch ein »Torrent«. Wählen Sie auf <http://www.tug.org/texlive/acquire-iso.html> den entsprechenden Link; das sollte das Übertragungsprogramm aufrufen, welches die Datei herunterlädt. Eine DVD kann man bekommen, indem man entweder einer $\text{T}_{\text{E}}\text{X}$ -Benutzer-Gruppe beitrifft oder indem man sie kauft: <http://www.tug.org/texlive/acquire-dvd.html>. Schließlich bindet ein Doppelklick auf das ISO-Abbild dieses als virtuelles Laufwerk in das System ein (im Fall einer DVD als echtes Laufwerk). Öffnen Sie nun eine Terminalsitzung und gehen Sie in das `texlive`-Verzeichnis des virtuellen (oder physikalischen) Laufwerks, etwa so:

```
$ cd /media/TeXLive2011
```

Anstelle von `»/media/TeXLive2011«` kann das Verzeichnis auch anders lauten; benutzen Sie die Tabulatortaste für die automatische Vervollständigung.

Installation der Distribution

Nun ist es Zeit den eigentlichen Installationsbefehl aufzurufen:

```
$ sudo ./install-tl -gui
```

Soll im Falle einer Online-Installation nicht der automatisch gewählte Spiegelserver verwendet werden, so kann über die Option `-repository` ein bestimmter Server angegeben werden, beispielsweise

```
$ sudo ./install-tl -gui \  
-repository=http://ctan.larsko.net/tex-archiv/systems/texlive/tlnet
```

Auf der Seite <http://mirror.ctan.org/CTAN.sites> sind alle aktuellen Spiegelserver gelistet.

Das System erfragt das Passwort und ein Fenster erscheint, ähnlich dem in Abbildung 1. Falls Sie als Standardpapiergröße Letter haben möchten, drücken Sie die entsprechende Schaltfläche »Ändern« (standardmäßig ist bei $\text{T}_{\text{E}}\text{X}$ Live A4 eingestellt). Am Fuß des Fensters gibt es eine Schaltfläche » $\text{T}_{\text{E}}\text{X}$ Live installieren«. Aktivieren Sie sie und seien Sie zuversichtlich, dass die Installation erfolgreich zum Abschluss kommt.

Ändern Sie auf keinen Fall die Standardeinstellung für »Symbolische Links in Systemverzeichnissen«: diese *muss* auf »Nein« lauten.

¹ Die Version `texlive2011-20110705.iso` ist 2431731712 Bytes groß (Anmerkung des Übersetzers)

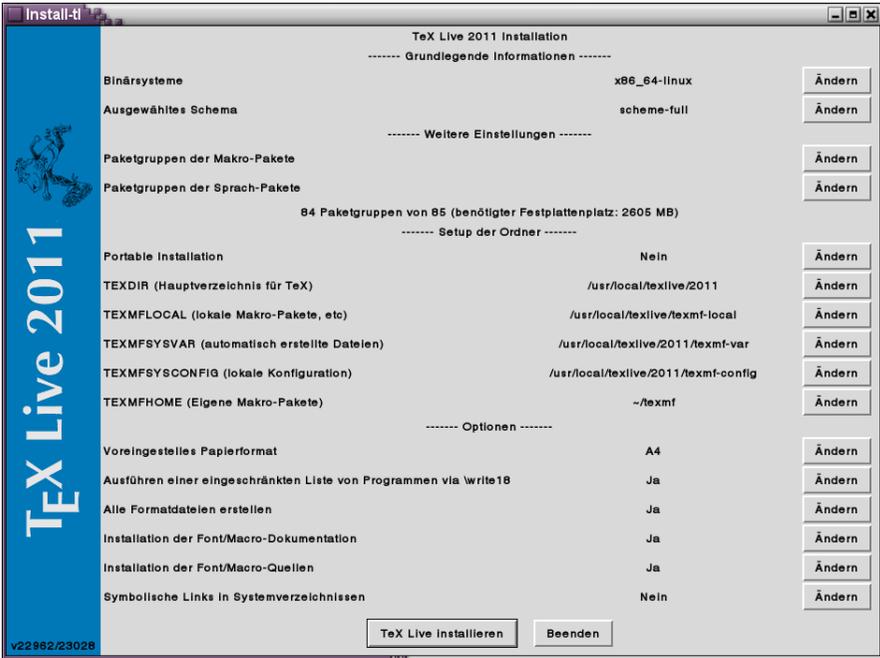


Abbildung 1: Grafische Oberfläche des Installationsbildschirms

Wurde die Installation erfolgreich beendet, gehen Sie zu Abschnitt »Nachbesserung der Installation«.

Wenn etwas schief geht

Traten bei der Installation Probleme auf, entfernt man vor dem nächsten Versuch am besten alles wieder, was bisher installiert wurde:

```
$ cd /usr/local/texlive
$ ls
 2011      texmf-local
$ sudo rm -rf 2011
$ cd -
```

Lassen Sie hier besondere Vorsicht walten, da der Befehl `rm` sehr gefährlich ist, da er ohne Rückfrage ganze Verzeichnisbäume löschen kann. Deshalb geben Sie die Befehle sehr sorgfältig ein. Die Ausgabe des zweiten Befehls (`ls`) sollte so aussehen wie angezeigt, außer dass vielleicht `texmf-local` fehlt. Bei einer

anderen Ausgabe kontrollieren Sie den ersten Befehl und geben ihn nochmals korrekt ein. Jetzt starten Sie den nächsten Installationsversuch.

Nachbesserung der Installation

Nun folgt der schwierige Teil: Machen Sie dem System bekannt, wo es die Programme der \TeX -Distribution findet. Indem man nichts an der originalen \TeX -Distribution von Ubuntu ändert, vermeidet man Probleme, wenn Programme wie etwa `kile` installiert werden, die davon abhängen. Es gibt eine Ausnahme, aber diese erfordert, dass man das `equivs`-Paket mit `Synaptic` installiert und einige sehr systemabhängige Tricks anwendet.

Gehen Sie nun zurück zum am Anfang erstellten Arbeitsverzeichnis

```
$ cd ~/texlive-install
```

und lassen Sie uns zum Betriebssystem übergehen. Folgende geheimnisvoll anmutenden Befehle

```
$ echo 'export PATH=/opt/texbin:${PATH}' > texlive.sh
$ sudo cp texlive.sh /etc/profile.d/
$ sudo mkdir -p /opt
```

erstellen eine Datei `texlive.sh`, die den Text enthält, den wir oben in der ersten Befehlszeile zwischen die einfachen Anführungszeichen geschrieben haben. Anschließend wird dabei diese Datei in ein Systemverzeichnis kopiert. Der nächste Schritt hängt nun von der Hardware-Architektur Ihres Computers ab; so verwenden Sie einen und nur einen der folgenden Befehle

```
$ sudo ln -s /usr/local/texlive/2011/bin/i386-linux /opt/texbin
$ sudo ln -s /usr/local/texlive/2011/bin/x86_64-linux /opt/texbin
$ sudo ln -s /usr/local/texlive/2011/bin/powerpc-linux /opt/texbin
```

Der Autor kann nicht wissen, welcher der richtige ist, die Wahl müssen Sie abhängig davon treffen, ob Ihr Computer einen 32-Bit-Intel/AMD-Prozessor, einen 64-Bit-x86-Prozessor oder einen PowerPC-Chip enthält. Herausfinden kann man das hier mit folgender Befehlszeile:

```
$ ls /usr/local/texlive/2011/bin
```

Als Antwort wird die benötigte Bezeichnung ausgegeben.

► Der erfahrene Benutzer mag vielleicht fragen, warum wir in `texlive.sh` nicht gleich den Namen des Verzeichnisses mit den ausführbaren Programmen eintragen, beispielsweise `export PATH=/usr/local/texlive/2011/bin/i386-linux:${PATH}`

Als Idee steckt dahinter, dass der Wechsel zur nächsten \TeX Live einfacher wird. Es reicht dann das Kommando

```
$ sudo ln -s /usr/local/texlive/2012/bin/i386-linux /opt/texbin
```

nach der Installation auszuführen *ohne, dass weitere Eingriffe nötig sind*. Beinahe ..., siehe später.

Melden Sie sich nun vom System ab und wieder an, damit diese Änderung wirksam wird. Öffnen Sie ein Terminalfenster und kontrollieren Sie, ob alles in Ordnung ist; der Befehl

```
$ which tex
```

sollte mit

```
/opt/texbin/tex
```

antworten. Wenn das der Fall ist, sind wir bereit, die Distribution auf den neuesten Stand zu bringen. Anderenfalls suchen Sie die Hilfe eines Gurus.

Es gibt zwei Wege, mit t_lm_gr zu arbeiten; am besten probieren Sie beide aus. Der erste startet t_lm_gr aus dem Terminal. Führen Sie den Befehl

```
$ gedit ~/.bashrc
```

aus und geben in dem sich öffnenden Fenster am Ende Folgendes ein:

```
...

# .bashrc addition for TeX Live
function sutlmgr () {
  if [[ -z "$@" ]]
  then
    sudo /opt/texbin/tlmgr -gui
  else
    sudo /opt/texbin/tlmgr "$@"
  fi
}
alias mktexlsr='sudo /opt/texbin/mktexlsr'
alias updmap-sys='sudo /opt/texbin/updmap-sys'
alias fmtutil-sys='sudo /opt/texbin/fmtutil-sys'
```

Die drei Punkte stehen für das, was bereits in der Datei `.bashrc` steht und keinesfalls verändert werden darf.

► Der erfahrene Benutzer mag vielleicht lieber die Datei `.bash_aliases` für diese Änderung bevorzugen.

Speichern Sie nun die so erweiterte Datei über den entsprechenden Menüeintrag, verlassen Sie `gedit` und geben im Terminalfenster folgende Befehle ein:

```
$ . ~/.bashrc
```

```
$ sutlmgr
```

Das `tlmgr`-Fenster erscheint, nachdem das System das Passwort erfragt hat. Von nun an wird der `sutlmgr`-Befehl das Verwaltungsprogramm `tlmgr` mit Administrator-Rechten ausführen. Ein Befehl wie

```
$ sutlmgr show --list xyz
```

ruft direkt die angeforderte `tlmgr`-Aktion auf. Mit

```
$ texdoc tlmgr
```

bekommt man die Dokumentation zu `tlmgr`.

Man kann auch eine kleine Anwendung für den Desktop erstellen. (Folgendes gilt für den Gnome-Fenstermanager; in KDE funktioniert das auf ähnliche Weise.) Begeben Sie sich mit der Maus auf die freie Desktop-Hintergrundfläche, drücken die rechte Maustaste und wählen »Create Launcher ...« (Anwendungsstarter erstellen). In dem sich nun zeigenden Fenster schreibt man als Namen »TeX Live Manager« (oder vielleicht eingedeutscht »TeX-Live-Verwaltungsprogramm«) in das Eingabefeld für den Namen und in das Eingabefeld für den Befehl (Command) kommt

```
gksu -d -S -D "TeX Live Manager" '/opt/texbin/tlmgr -gui'
```

Nach dem Anlegen der Anwendung erfragt ein Doppelklick auf das Symbol das Passwort und startet `tlmgr`.

Eine letzte Sache bleibt noch zu tun: Teilen Sie dem Betriebssystem die OpenType-Fonts mit, die die T_EX Live mitbringt, um diese in X_Y(L)_AT_EX mittels Schriftnamen und nicht nur via Dateinamen auswählen zu können.

```
$ sudo cp $(kpsewhich -var-value TEXMFSYSVAR)\
  /fonts/conf/texlive-fontconfig.conf /etc/fonts/conf.d/09-texlive.conf
$ sudo fc-cache -fsv
```

Eine tabellarische Zusammenfassung der Befehle all dieser Arbeitsschritte bei einer typischen Ubuntu-10-Installation finden Sie auf Seite 45; in kursiver Schrift stehen alle Teile, die unterschiedlich sein können; die unterstrichenen Zeilen geben die Antworten des Systems wieder. Die kursive und unterstrichene Antwort verrät, was anstelle von `i386-linux` in der folgenden Zeile eingesetzt werden muss; Zeilen mit Anweisungen zwischen spitzen Klammern beschreiben Dinge, die außerhalb des Terminals erledigt werden müssen.

OpenSUSE

Die beschriebene Vorgehensweise funktioniert mit OpenSUSE, vorausgesetzt, dass Perl-Tk installiert wurde. Man braucht keine Funktion zu definieren, um `tlmgr` zu starten, aber man muss sich dann die Optionen `-c` für `su` und `-E` für `sudo` merken. Die Verwaltungsprogramme lassen sich somit folgendermaßen aufrufen:

```
$ su -c tlmgr -gui
$ sudo -E updmap-sys
$ sudo -E mktexlsr
```

Fedora

Dasselbe Verfahren lässt sich auch für Fedora zumindest in der Version 13 verwenden. Das Perl-Tk-Modul fehlt hier in der Standard-Distribution von Fedora. Man erhält es unter

<http://koji.fedoraproject.org/koji/buildinfo?buildID=151517>

Die Datei `/etc/profile.d/texlive.sh` enthält einen anderen Text:

```
#!/bin/bash
if ! echo $PATH | /bin/egrep -q "(^|:)/opt/texbin($|:)"
then
    PATH=/opt/texbin:$PATH
fi
```

Unterschiedlich ist auch, dass in Fedora nur root als Administrator dient. Im Gegensatz zur Standard-Konfiguration in Ubuntu erfordert sudo daher das Administrator-Passwort.

► Anstelle von sudo kann man auch mit su arbeiten, das eine Shell mit Administrator-Rechten öffnet. Dann gibt man die angegebenen Befehle ohne vorangestelltes sudo ein. Anschließend verlässt man die Shell wieder mit exit. Man beachte, dass innerhalb von su die Tilde nicht mehr das Home-Verzeichnis des Benutzers referenziert, der su aufgerufen hat, sondern das des Administrators root.

Übergang zu T_EX Live 2012

Wenn dann irgendwann die nächste Ausgabe der T_EX Live fertig wird, installiert man diese genauso, wie das in den vorigen Abschnitten zur Beschaffung und Installation der Distribution angegeben wird.² Danach führt man noch folgende Befehle aus, wobei Sie `i386-linux` durch Ihr Verzeichnis ersetzen:

```
$ sudo ln -s /usr/local/texlive/2012/bin/i386-linux /opt/texbin
$ sudo cp $(kpsewhich -var-value TEXMFSYSVAR) \
    /fonts/conf/texlive-fontconfig.conf/etc/fonts/conf.d/09-texlive.conf
$ sudo fc-cache -fsv
```

Das war alles (natürlich passt man wieder `i386-linux` entsprechend seiner Architektur an.) Das Erstellen der Datei `texlive.sh` entfällt ebenso wie das Ab- und Anmelden am System. So sind wir auf T_EX Live 2012 bestens vorbereitet.

²Die Jahresangabe aktualisiert man entsprechend auf 2012 (Anmerkung des Übersetzers).

Anhang

In diesem Anhang gehen wir davon aus, dass T_EX Live auf einem Ubuntu-System wie in der hier umrissenen Art und Weise installiert wurde. Ändern Sie die Befehlszeilen bezüglich `tlmgr`, wenn Sie mit einer anderen Distribution arbeiten.

Installation eines privaten Pakets

Lassen Sie uns annehmen, dass wir ein L^AT_EX-Paket brauchen, das es nicht auf T_EX Live gibt; sei es aus Lizenzgründen oder weil wir eine experimentelle Version ausprobieren wollen. Es gibt zwei Orte, in die die nötigen Dateien gesteckt werden können. Zuerst einmal laden Sie das Archiv von wo auch immer (CTAN oder irgendwo anders) herunter und packen es in einem Arbeitsverzeichnis aus. Um mal ein Beispiel zu geben, das Paket heiße `padua` und das Verzeichnis enthalte die Dateien `README`, `padua.ins`, `padua.dtx` und `padua.pdf` (die Teile in Kursivschrift stehen für den echten Namen). Öffnen Sie ein Terminalfenster und schreiben dort folgenden Befehl:

```
$ tex padua.ins
```

Wenn es die Datei mit der Endung `.ins` nicht geben sollte, dann wird vermutlich

```
$ tex padua.dtx
```

dasselbe leisten. In beiden Fällen wird das System einige Dateien erzeugen, die noch an die richtige Stelle platziert werden müssen. Natürlich ist das ein einfaches und allgemein gehaltenes Beispiel; einige Pakete haben eine kompliziertere Struktur und die Anweisungen des Autors sollten befolgt werden, angepasst an das, was hier steht.

Jetzt müssen Sie sich entscheiden, ob Sie einen *privaten* oder einen *lokalen* Baum benutzen wollen. Der Hauptunterschied ist, dass im zweiten Fall alle Benutzer des Computers das Paket benutzen können; dafür werden Administrator-Rechte benötigt, um den lokalen Baum bearbeiten zu können. Das Wort »Baum« bezieht sich auf eine hierarchische Verzeichnisstruktur, speziell darauf zugeschnitten, dass das T_EX-System effizient damit arbeiten kann.

In Linux-Systemen hat der private Baum seinen Ausgangspunkt üblicherweise in `~/texmf`. Es ist ein Unterverzeichnis Ihres Heimatverzeichnisses. Der lokale Baum liegt in `/usr/local/texlive/texmf-local`. Tatsächlich ist es gar nicht nötig zu wissen, wo sie genau liegen. Wir können uns nämlich ein Kürzel definieren, das für den Ort des gewählten Baumes steht; für den privaten Baum erledigt das

```
$ Local=$(kpsewhich -var-value TEXMFHOME)
```

und für den lokalen Baum führen wir

```
$ Local=$(kpsewhich -var-value TEXMFLocal)
```

aus. Das T_EX-System ist so konfiguriert, dass es die Orte selber kennt. Der Zaubertick dieser Befehlszeile liegt nun darin, dass dieser Ort ermittelt und der Variablen `Local` zugewiesen wird.

Lassen Sie uns nun da weitermachen, wo wir aufgehört haben. Wir legen die nötigen Verzeichnisse an und kopieren die Dateien hinein.

```
$ mkdir -p $Local/source/latex/padua
$ cp README padua.ins padua.dtx $Local/source/latex/padua
$ mkdir -p $Local/doc/latex/padua
$ cp padua.pdf $Local/doc/latex/padua
$ mkdir -p $Local/tex/latex/padua
$ cp *.sty ... $Local/tex/latex/padua
```

In der letzten Zeile stehen die drei Punkte für weitere Dateien, die von (L^A)T_EX eingelesen werden können sollten und typischerweise aus der `.dtx`-Datei erzeugt worden sind.

Installation einer Schriftfamilie

Weltweit gibt es zahlreiche Anleitungen, wie man neue Schriften installiert, die man gekauft hat oder frei herunterladen konnte. Nicht zu empfehlen ist es, die Schriften im privaten Baum zu installieren, da dies ständige Arbeit des Benutzers verlangt, wenn Aktualisierungen der T_EX Live-Distribution die Schriften betreffen.

Am besten folgt man der Anleitung in der Broschüre »The font installation guide« von Philipp Lehmann, die in T_EX Live mit folgender Befehlszeile aufrufbar ist:

```
$ texdoc fontinstallationguide
```

Man folge diesen Anweisungen, bis die benötigten Dateien erstellt sind. Wir nehmen an, die Schriftfamilie heiße »Padua« und der T_EX-Familiename sei `zpd`. Wie gewöhnlich bezeichnet die kursive Schrift die variablen Namensteile. Nach Lehmanns Anleitung generiert man eine ganze Reihe von Dateien im Arbeitsverzeichnis, die verschiedene Endungen haben:

```
.tfm .vf .pfb .afm .map .sty .fd
```

Diese Dateien müssen an die korrekten Plätze in der T_EX-Verzeichnis-Hierarchie gelegt werden. Der richtige Ort ist der bereits beschriebene *lokale Baum*. Wiederum brauchen wir nicht zu wissen, wo er genau liegt; lassen Sie uns ein Kürzel definieren, die Verzeichnisstruktur anlegen und die Dateien kopieren.

```
$ Local=$(kpsewhich -var-value TEXMFLOCAL)
$ sudo mkdir -p $Local/fonts/{afm,tfm,type1,vf}/padua
$ sudo cp zpd*.afm $Local/fonts/afm/padua
$ sudo cp zpd*.tfm $Local/fonts/tfm/padua
$ sudo cp zpd*.pfb $Local/fonts/type1/padua
$ sudo cp zpd*.vf $Local/fonts/vf/padua
```

```
$ sudo mkdir -p $Local/tex/latex/padua
$ sudo cp *.sty *.fd $Local/tex/latex/padua
$ sudo mkdir -p $Local/fonts/map/dvips/padua
$ sudo cp padua.map $Local/fonts/map/dvips/padua
$ mktexlsr
```

Wir haben die Möbel aufgestellt, nun müssen wir noch dem \TeX -System die Türschlüssel geben. Es gibt zwei Fälle: Es ist das erste Mal, dass wir eine neue Schriftfamilie installieren oder wir haben dies bereits schon einmal gemacht. Im ersten Fall müssen wir eine Datei erzeugen und an einer geeigneten Stelle ablegen:

```
$ echo "Map padua.map" > updmap-local.cfg
$ mkdir -p $Local/web2c
$ sudo mv updmap-local.cfg $Local/web2c
$ sutils generate --rebuild-sys updmap
```

Im zweiten Fall hängen wir nur eine Zeile an eine bereits bestehende Datei an:

```
$ cp $Local/web2c/updmap-local.cfg .
$ echo "Map padua.map" >> updmap-local.cfg
$ sudo mv updmap-local.cfg $Local/web2c
$ sutils generate --rebuild-sys updmap
```

(Die letzte Aktion kann man genauso wie den Aufruf von `mktexlsr` von der graphischen Benutzerschnittstelle von `tlmgr` aus ausführen.) Auf diesem Weg können wir sicher sein, dass die Türschlüssel nicht mit \TeX Live-Aktualisierungen verloren gehen. Siehe auch <http://www.tug.org/fonts/fontinstall.html> für eine ausführlichere Beschreibung dieser Schritte. Falls es zufällig auch die OpenType-Versionen von unseren Schriften gibt, dann fügen Sie folgende Befehlszeilen zu den entsprechenden, die wir zuvor gesehen haben:

```
$ sudo mkdir -p $Local/fonts/opentype/padua
$ sudo cp *.otf $Local/fonts/opentype/padua
```

Ersetzen Sie analog `opentype` durch `trueType`, wenn die Fonts die Endung `.ttf` haben. Wenn Sie das erste Mal OpenType-, TrueType- oder Type-1-Schriften dem lokalen Baum hinzufügen, müssen Sie dies dem System bekannt machen:

```
$ cp /etc/fonts/conf.d/09-texlive.conf 09-texlive-local.conf
$ gedit 09-texlive-local.conf
```

Ändern Sie im Fenster, das nun erscheint, alle Zeichenketten »2011/texmf-dist« in »texmf-local«. Speichern und arbeiten Sie im Terminal weiter:

```
$ sudo mv 09-texlive-local.conf /etc/font/conf.d
$ sudo fc-cache -fsv
```

Nun kann auch \LaTeX auf die neuen Schriften zugreifen. Sie brauchen nur noch `fc-cache` aufzurufen, wenn die »lokale« Konfigurationsdatei erstellt worden ist.

Tabelle 1: Installationszusammenfassung für Ubuntu

```

<Installiere Perl-Tk und Perl-doc mit Synaptic>
<Öffne ein Terminalfenster >
$ mkdir ~/texlive-install
$ cd ~/texlive-install
$ wget http://mirror.ctan.org/systems/texlive/tlnet/install-tl-unx.tar.gz
$ tar xzf install-tl-unx.tar.gz
$ cd install-tl-20110705
$ sudo ./install-tl -gui -repository http://mirror.ctan.org/systems/texlive/tlnet
<Drücke "Ändern" für das Papierformat, falls es geändert werden soll>
<Drücke "Installiere TeX Live">
<Warte, bis die Installation fertig ist; trinke einen Kaffee oder vielleicht auch zwei>
<Drücke "Ende">
$ cd ~/texlive-install
$ echo 'export PATH=/opt/texbin:${PATH}' > texlive.sh
$ sudo cp texlive.sh /etc/profile.d/
$ sudo mkdir -p /opt
$ ls /usr/local/texlive/2011/bin
  i386-linux
$ sudo ln -s /usr/local/texlive/2011/bin/i386-linux /opt/texbin
<Melde dich vom System ab>
<Nach dem Anmelden öffne ein Terminal>
$ which tex
  /opt/texbin/tex
<Wenn die Antwort anders ausfällt, schrei um Hilfe>
$ gedit ~/.bashrc
<Hänge ans Ende der Datei>
# Additions for TeX Live
function sutlmgr () {
  if [[ -z "$@" ]]
  then
    sudo /opt/texbin/tlmgr -gui
  else
    sudo /opt/texbin/tlmgr "$@"
  fi
}
alias mktexlsr='sudo /opt/texbin/mktexlsr'
alias updmap-sys='sudo /opt/texbin/updmap-sys'
alias fmtutil-sys='sudo /opt/texbin/fmtutil-sys'
<Sichere und verlasse gedit>
$ sudo cp $(kpsewhich -var-value TEXMFSYSVAR)/fonts/conf/texlive-fontconfig.conf \
  /etc/fonts/conf.d/09-texlive.conf
$ sudo fc-cache -fsv
<Entspanne dich und genieße  $\TeX$  Live 2011>

```

Anmerkungen.

(1) Das Datum 20110705 ist ein Beispiel und kann mittlerweile anders sein.

(2) i386-linux bezieht sich auf eine der möglichen Rechnerarchitekturen (auch Plattformen genannt); hier kann es sich auch um x86_64-linux oder weniger wahrscheinlich um powerpc-linux handeln.

Installation nicht ganz freier Fonts

Reinhard Kotucha

Während der »Bauarbeiten« an T_EX Live 2005 sind einige Type-1-Fonts entdeckt worden, die aus Lizenzgründen aus der Distribution entfernt werden mussten. Zwar dürfen diese Fonts frei verwendet werden, aber für deren Distribution darf kein Geld verlangt werden. Letzteres ist aber bei Verbreitung auf CD oder DVD nicht vermeidbar.

Da diese Zeichensätze irrtümlich in früheren Versionen von T_EX Live vorhanden waren, war davon auszugehen, dass sie bereits in einigen Dokumenten verwendet wurden. Während einer Diskussion mit Karl Berry kam die Idee auf, diese Fonts per Netzwerk-Installation verfügbar zu machen.

Manuelle Installation

Prinzipiell ist es nicht sehr schwierig, einen Zeichensatz, für den die von T_EX/L^AT_EX benötigten Dateien bereits vorhanden sind, zu installieren. Es gibt sehr viele solcher Pakete auf CTAN. Üblicherweise reicht es aus, die entsprechende .zip-Datei in der Wurzel eines texmf-Baums auszupacken und, wie in der Installationsanleitung beschrieben, updmap aufzurufen.

Das funktioniert recht gut, wenn die Verzeichnisstruktur innerhalb der .zip-Datei TDS¹-konform ist. Bei nicht freien Fonts ist das aber häufig nicht der Fall und Dateien müssen mühsam in die entsprechenden Verzeichnisse einsortiert und gegebenenfalls umbenannt werden.

Automatische Installation mit getnonfreefonts

Erster Anlauf: ein Shell-Script für Unix

Ein Shell-Skript, das die Dateien von den entsprechenden Servern herunterlädt und installiert, war schnell geschrieben. Da aber alles sehr kurz vor der Veröffentlichung von T_EX Live 2005 entstand, blieb keine Zeit mehr, das Programm überhaupt zu testen. Daher wurde in letzter Minute eine recht unkonventionelle Maßnahme ergriffen: Das Programm wurde kurzerhand in zwei Teile aufgeteilt. Ein kleiner Teil ging auf die CD, der Teil, der die eigentliche Arbeit verrichtet, auf den Server der TUG. Damit war es möglich, auch später noch Fehler zu korrigieren oder auf Änderungen auf CTAN zu reagieren.

¹ T_EX Directory Standard, <http://tug.org/tds/tds.html>

T_EX Live 2005 hat die Programme `getnonfreefonts` und `getnonfreefonts -sys` mitgeliefert, die, um Fonts zu installieren, das Programm `getfont` herunterladen und ausführen.

`getnonfreefonts` für Windows

Als Shell-Skript war das Programm nur unter Unix ausführbar. Um es auch für Windows bereitzustellen, wurde es für T_EX Live 2007 in Perl neu implementiert.

T_EX Live ohne `getnonfreefonts`

Damit T_EX Live auch in Linux-Distributionen verwendet werden kann, die keine Programme akzeptieren, mit denen man unfreie Software installieren kann, ist `getnonfreefonts` vor etwa zwei Jahren aus T_EX Live entfernt worden und muss nun separat installiert werden.

Installation

Um das Programm mit neueren T_EX Live-Versionen als 2009 benutzen zu können, muss es zunächst installiert werden. Der Installer wird dazu von

```
http://tug.org/fonts/getnonfreefonts
```

in einem beliebigen Verzeichnis heruntergeladen. Danach ruft man in diesem Verzeichnis

```
texlua install-getnonfreefonts
```

auf. Wenn man T_EX Live für mehrere Betriebssysteme installiert hat, sollte man den letzten Schritt *nicht* unter Windows ausführen, da dann die für Unix erforderlichen symbolischen Links nicht erzeugt werden können.

Wer T_EX Live als Administrator installiert hat, muss auch `getnonfreefonts` mit Administratorrechten installieren.

Benutzt man ein T_EX Live, das Bestandteil einer Linux-Distribution ist, sollte man zunächst überprüfen, ob `getnonfreefonts` vorhanden ist. Wenn nicht, kann man den Installer ausprobieren. Linux-Distributionen haben eine etwas andere Verzeichnisstruktur als das »offizielle« T_EX Live. Der Installer berücksichtigt das und installiert das Programm in `/usr/local/bin` und die anderen Dateien in `$TEXMFLOCAL` anstatt im von der Linux-Distribution verwalteten T_EX Live-Verzeichnisbaum. Damit werden Konflikte mit dem Paket-Manager vermieden.

Installation der Zeichensätze

Nach erfolgreicher Installation stehen, wie von T_EX Live gewohnt, zwei Programme zur Verfügung, `getnonfreefonts` und `getnonfreefonts-sys`.

Das Ertere installiert die Fonts für den aktuellen Benutzer im Verzeichnis `$TEXMFHOME`, das Letztere für alle Benutzer in `$TEXMFLOCAL`. Empfohlen wird die Verwendung von `getnonfreefonts-sys`, falls man die dafür notwendigen Zugriffsrechte hat.

Mit `getnonfreefonts[-sys] --help` erhält man eine kurze Hilfe, in der auch angezeigt wird, in welchem Verzeichnisbaum die Fonts installiert werden. Eine etwas ausführlichere Dokumentation bekommt man mit `man getnonfreefonts` oder `texdoc getnonfreefonts`. Mit der Option `--lsfonts` werden alle verfügbaren Fonts aufgelistet. Die zur Zeit vorhandenen Schriften sind in Tabelle 1 zusammengefasst.

Tabelle 1: Zusammenstellung der Fonts

<i>Paketname</i>	<i>Name</i>	<i>Hersteller</i>
<code>arial-urw</code>	Arial	URW
<code>classico</code>	Classico	URW
<code>dayroman</code>	DayRoman	Apostrophiclabs
<code>eurofont</code>	Euro symbols	Adobe
<code>garamond</code>	GaramondNo8	URW
<code>lettergothic</code>	LetterGothic	URW
<code>luximono</code>	LuxiMono	Bigelow & Holmes
<code>vntex-nonfree</code>	VnT _E X nonfree	Hàn Thê Thàn
<code>webomints</code>	Webomints	Galapagos Design Group

Das Kommando

```
getnonfreefonts-sys garamond classico
```

installiert GaramondNo8 und Classico in `$TEXMFLOCAL`. Anstatt alle Fonts aufzulisten, kann man auch die Option `--all` benutzen. Die Option `--http` kann verwendet werden, wenn man sich hinter einer Firewall befindet, die FTP-Verkehr verhindert. In diesem Fall hat man keinen Zugriff auf die Euro-Fonts von Adobe.

Bei der Installation einer neuen Version von T_EX Live bleiben zwar Dateien in `$TEXMFHOME` und `$TEXMFLOCAL` unberührt, allerdings gehen die entsprechenden Einträge in `updmap.cfg` verloren. Mit der Option `--refreshmap` können die Einträge wieder hergestellt werden.

Lizenzen

Die von `getnonfreefonts` zur Verfügung gestellten Zeichensätze dürfen zwar frei benutzt werden, die Weitergabe unterliegt allerdings Einschränkungen. Bei kommerzieller Nutzung oder Modifikationen wird empfohlen, die Lizenzen genau zu studieren.

Weitere Zeichensätze

Gelegentlich taucht die Frage auf, ob nicht der eine oder andere Zeichensatz auch von `getnonfreefonts` unterstützt werden kann. Dazu müssen folgende Bedingungen erfüllt sein:

- Der Font muss frei benutzbar sein, auch im kommerziellen Umfeld.
- Die für \TeX benötigten Metrik-Dateien müssen verfügbar sein, vorzugsweise auf CTAN.
- Der Zeichensatz sollte, zumindest für eine bestimmte Sprache, vollständig sein, und eine für die Verwendung mit \TeX angemessene Qualität aufweisen.
- Alle benötigten Dateien müssen in `.zip`-Archiven von Servern bereitgestellt werden, die auch langfristig zuverlässig verfügbar sind, vorzugsweise CTAN.

Auch wenn der Name des Programms anderes vermuten lässt: `getnonfreefonts` dient nicht dazu, Lizenzen zu umgehen. Lediglich Zeichensätze, für deren Weitergabe kein Geld verlangt werden darf, deren Lizenz ansonsten für \TeX Live angemessen ist, können unterstützt werden.

Die experimentellen Trennmuster für \LaTeX

Rolf Niepraschk

Einleitung

Anfang Juli kündigte Stephan Hennig eine neue Version des Paketes »dehyph-exptl«, der noch immer als »experimentell« bezeichneten deutschen Trennmuster, an:

2011-07-02: v0.22

Dieses Paket enthält experimentelle Trennmuster für die deutsche Sprache. Die Trennmuster decken das in Deutschland, Österreich und der Schweiz gebräuchliche Standarddeutsch

in der traditionellen und reformierten Rechtschreibung ab und können mit den Paketen `babel` und `hyphsubst` aus dem Oberdiek-Bündel verwendet werden.

- *Etwa 200 Einträge wurden korrigiert.*
[...]
- *Die Dokumentation wurde erweitert.*
[...]

<http://mirror.ctan.org/language/hyphenation/dehyph-exptl>

<http://www.ctan.org/pkg/dehyph-exptl>

Im Folgenden soll gezeigt werden, wie die neuen Trennmuster in einem \LaTeX -Dokument verwendet werden können. Zu Hintergründen und weitergehenden Fragen sollten die in deutscher Sprache verfassten Dokumentationen zu Rate gezogen werden. [3,4]

Die Anwendung

Da die verbreiteten \TeX -Distributionen bei der Erzeugung der \LaTeX -Formate nach wie vor relativ alte Trennmuster verwenden, besteht der einfachste Weg darin, diese nachträglich zu ersetzen. Zu diesem Zweck ist das Paket `hyphsubst` geschrieben worden (siehe: [2]). Das folgende kurze \LaTeX -Dokument zeigt eine Möglichkeit, sowohl die Trennmuster der alten als auch der neuen Rechtschreibung auf den durch die »experimentellen Trennmuster« repräsentierten neuesten Stand zu bringen. Sollten diese nicht in der \TeX -Installation enthalten und im Format eingebunden sein, gibt es keine Fehlermeldung. Es wird dann wie bisher verfahren.

```
\documentclass[fontsize=11pt,paper=a4,pagesize]{scrartcl}
\usepackage[utf8]{inputenc}
\usepackage[T1]{fontenc}
\usepackage{hyphsubst}
\HyphSubstIfExists{ngerman-x-latest}{%
  \HyphSubstLet{ngerman}{ngerman-x-latest}}{}
\HyphSubstIfExists{german-x-latest}{%
  \HyphSubstLet{german}{german-x-latest}}{}
\usepackage{ngerman}{babel}
\begin{document}
Hallo Welt!
\end{document}
```

Dass die verbesserten Trennmuster verwendet werden, erkennt man daran, dass in der log-Datei Angaben ähnlich der folgenden zu finden sind:

```
Package hyphsubst Info: Redefined: \l@ngerman
(hyphsubst)          old value: 29
(hyphsubst)          new value: 4 on input line 6.
```

```
Package hyphsubst Info: Redefined: \@german
(hyphsubst)          old value: 28
(hyphsubst)          new value: 3 on input line 8.
```

Die Bezeichnung `ngerman-x-latest` ist ein Alias für die jeweils neueste Version der Trennmuster. Da die Trennmuster weiterentwickelt werden, ist es nicht ganz ausgeschlossen, dass eine neuere Version zwar Verbesserungen aufweist, jedoch gleichzeitig dazu führen kann, dass Trennungen, die in der Vorversion korrekt waren, nun fehlerhaft sind. Auch wenn die Wahrscheinlichkeit dafür nicht groß ist, sollte man bei einem Dokument, bei dem man solche Effekte unbedingt ausschließen möchte, statt `ngerman-x-latest` (bzw. `german-x-latest`) besser `ngerman-x-YYYY-mm-dd` (bzw. `german-x-YYYY-mm-dd`) verwenden. Statt »YYYY-mm-dd« ist das zur Bearbeitungszeit des Dokuments aktuelle Versionsdatum der Trennmuster zu verwenden. Man kann dieses folgendermaßen ermitteln (nach einem Tipp von Stephan Hennig in der news group »de.comp.text.tex«):

```
grep -i german `kpsewhich language.dat`
```

(unter einem Unix-artigen Betriebssystem)

```
for /F "usebackq" %f in (`kpsewhich language.dat`) do find /i "german" %f
```

(unter Microsoft Windows)

Fazit

Die Verwendung der im Paket »dehyph-exptl« enthaltenen »experimentellen Trennmuster« in einem deutschsprachigen \LaTeX -Dokument ist sehr zu empfehlen. Es werden danach nahezu keine Wörter mehr falsch getrennt, was einen selbstverständlich immer noch nicht von der Kontrolle dieser Trennungen entbindet. Es werden nun mehr und bessere Trennstellen gefunden, was zu einer insgesamt erheblich verbesserten Qualität des endgültigen Dokumentes führt. Es soll an dieser Stelle den Autoren des Paketes, also der »deutschsprachigen Trennmustermanschaft«, herzlich für ihre hervorragende Arbeit gedankt werden. Da mittlerweile das Attribut »experimentell« durchaus durch »sehr stabil« ersetzt werden könnte, bleibt die Hoffnung, dass möglichst bald diese Trennmuster standardmäßig in die \LaTeX -Formate eingearbeitet werden, was das nachträgliche Ersetzen, wie hier beschrieben, unnötig werden ließe.

Literatur und Software

- [1] DANTE e. V.: *Trennmuster-Wiki*; <http://projekte.dante.de/Trennmuster/WebHome>.
- [2] Heiko Oberdiek: *The hyphsubst package*; CTAN:tex-archive/macros/latex/contrib/oberdiek/hyphsubst.pdf; Juni 2008.

- [3] Die deutschsprachige Trennmustermannschaft: *Experimentelle Trennmuster für die deutsche Sprache*; CTAN: Language/hyphenation/dehyph-exptl/dehyph-exptl.pdf; Juli 2011.
- [4] Die deutschsprachige Trennmustermannschaft: *Freie Wortlisten und Trennmuster für die deutsche Sprache. Eine kurze Projektbeschreibung*; 2011.

Latin Modern Math

Herbert Voß

Über mathematische Schriftsätze wurde schon mehr als einmal in dieser Zeitschrift berichtet. Die einzigen halbwegs kompletten Zeichensätze, die für eine Nutzung unter pdf^lTeX sowohl Text- als auch zugehörige mathematische Zeichen enthalten, sind Computer Modern und kp-Fonts, allerdings beschränkt auf 256 Zeichen pro Schrift. Alle anderen mathematischen Zeichensätze, die zu häufig benutzen Textschriften passen, sind nicht im Bereich Open Source angesiedelt, beispielsweise Times-Roman, Helvetica, Lucida oder Palatino. Die Revision der Computer Modern führte zur Latin Modern, für die jetzt ebenfalls ein mathematischer Zeichensatz vorliegt, die Latin Modern Math im OpenType-Format (.otf). Sie ist Bestandteil von TeX Live 2011 und vervollständigt damit die bisher schon verfügbaren Glyphen der Schriftfamilie Latin Modern. Zudem kann die Latin Modern Math durch die interne Anordnung der mathematischen Zeichen gleichermaßen von XeTeX, LuaTeX, LibreOffice und auch MS-Word eingesetzt werden. Eine entsprechende Mitteilung gab es von der GUST, die seit Jahren die Überarbeitung der Computer Modern und der Standardschriften von PostScript übernommen hat (deutsche Übersetzung):

Liebe TeX-Gyre-Freunde und Interessierte,

mit großem Vergnügen kündigt ich an – keine Neuigkeiten für einige von euch :-), dass seit Anfang Juli die Beta-Version 0.903 der Latin-Modern-Math-Fonts im OpenType-Format auf der GUST-Webseite erhältlich ist: <http://www.gust.org.pl/projects/e-foundry/lm-math>

Obwohl die Fonts sich noch im Beta-Stadium befinden, sind sie bereits ziemlich gut benutzbar – eine erste schnelle, aber ausführliche Prüfung, durchgeführt von Ulrik Vieth und anderen (danke an alle), zeigte keinen ernsten Probleme. Zum Zeitpunkt dieses Schreibens gibt es noch kein vollständiges Release-Paket, beispielsweise fehlen die GFL-Lizenzinformation (GUST Font License) genauso wie Readme-Dateien und Angaben zur Versionsgeschichte. LM Math schaffte es in die TeX Live 2011, wo sie zu den normalen LM-Fonts gesteckt

wurde. (Das ist nicht die bevorzugte Weise, sondern nur eine Behelfslösung in letzter Minute). Aber die Mathematikschrift ist noch nicht auf CTAN verfügbar – eben weil das Release-Paket noch nicht fertig wurde. Ich denke, die Veröffentlichung der LM Math ist aus verschiedenen Gründen bedeutend:

1. Sie vervollständigt die »Latin-Modern-isierung« der Computer-Modern-Fonts – es ist nun möglich, »echte« mathematische Texte in fast jeder Lateinschrift zu setzen, ohne Schriftarten außerhalb von LM und LM Math zu benötigen.
2. Sie ist der vierte OTF-Mathematik-Font, der jemals veröffentlicht wurde – nach Cambria Math, Asana Math und XITS (wenngleich es Gerüchte über die Entwicklung von Lucida Math und Euler Math gibt). So bleibt die T_EX-Welt eine der größten Akteure bei Werkzeugen und Material für mathematischen Textsatz.
3. Obwohl man es ohne etwas Insiderkenntniss nicht sofort sieht, das e-foundry-Team verfügt nun über eine einsetzbare Werkzeug-Kette, um mathematische Fonts im OpenType-Format zu erstellen; viele kleine und nicht so kleine Fallen (das Team nennt sie halb liebevoll Bestien (im Englischen »Beasts« mit großem »B«) wurden erkannt und entschärft oder zumindest etwas gezähmt.

Der letzte Grund ist sicher der wichtigste, zumindest aus Sicht des T_EX-Gyre-Math-Projekts. Das Team ist nun in der Lage, neue Mathematikschriften auf nahezu automatische Art zu erstellen: Man muss sich lediglich um Gestaltdetails der Mathematikzeichen kümmern, die sich auf die Grundtextschriftart beziehen. Das heißt nicht, dass das Aussehen der mathematischen Zeichen unwichtig wäre – im Gegenteil natürlich. Der Fortschritt, der hier erreicht wurde, bedeutet, dass jene Probleme als Hauptprobleme mit neuen mathematischen Fonts übrigbleiben, so dass das Team sich auf diese konzentrieren kann.

Ergebnis: Es sieht so aus (man klopfte auf Holz!), dass Mathematikfonts für die Textschriftarten der T_EX-Gyre-Sammlung nicht weit entfernt sind ... drückt die Daumen!

Viel Spass mit LM Math!

Jerzy Ludwiczowski

(für die GUST e-foundry)

Die folgende Zusammenstellung zeigt einige der verfügbaren Zeichen der neuen Mathematikschrift, wobei die hexadezimale Angabe der Position im Unicode nur eine Hilfe sein soll, wo man dieses Zeichen *ungefähr* findet. Innerhalb einer Tabellenzeile ist die Schrittweite dieser Nummer nicht zwingend gleich Eins.

Die Erstellung der Tabelle erfolgte mit Lua^LT_EX, wobei die Lua-Funktion im Wesentlichen auf der Idee von Philipp Stephani beruht. Es werden insgesamt acht Spalten für die Zeichen reserviert, die dann über eine Schleife in Lua gefüllt werden. Da die Reihenfolge des Einlesens nicht der numerischen Reihenfolge der Zeichen entspricht, wird die Liste noch sortiert.

```
\font\lmmath={file:lmmath-regular.otf:mode=base} at 10pt\relax
\begin{luacode*}
```

```

-- nach einer Idee von Philipp Stephani
function print_glyphs(maxCols,maxChars) -- Anzahl Spalten und Zeichen
  local id = font.id("lmmath") -- Font ID holen
  local fnt = font.getfont(id)
  local col = 1
  local maxU4 = 15*(16^3+16^2+16+1)
  a = {}
  for k, v in pairs(fnt.characters) do
    a [#a + 1] = k
  end
  table.sort(a)
  for i, k in ipairs(a) do
    if i >= maxChars then break end
    if col == 1 then
      if k >maxU4 then
        tex.sprint(string.format("U+%06x", k))
      else
        tex.sprint(string.format("U+%04x", k))
      end
      tex.sprint("&")
    end
    tex.sprint(string.format([[\\char%i]], k))
    if col == maxCols then -- Zeile voll?
      tex.sprint([[\\cline{2-} .. maxCols+1 .. "} "]) -- ja, also abschließen
      col = 1 -- und neu starten
    else
      tex.sprint("&") -- nein, also & ausgeben
      col = col + 1 -- Spalte inkrementieren
    end
  end
end
\end{luacode*}

```

Die eigentliche Tabellendefinition besteht dann nur aus wenigen Zeilen. Vor der Tabelle wird mit `\lmmath` die Schrift ausgewählt und in der Tabelle wird mit `\directlua` die Lua-Funktion `print_glyphs` (Spalten, Zeichen) aufgerufen, die dann das Auffüllen der Tabelle übernimmt.

```

\color{black!20}\lmmath
\begin{longtable}{>{\color{black!50}\ttfamily\footnotesize}r|
                 *{10}{>{\color{black}}p{1.5em}}|}
\caption{Tabellarische Anordnung der ersten 1360 Zeichen der LM Math}\
\cline{2-11}
\endfirsthead
\cline{2-11}
\endhead
\directlua{print_glyphs(10,1360)} \ \ \cline{2-11}
\end{longtable}

```

Die Glyphen des Fonts werden von LuaTeX nicht zwingend in der numerischen Reihenfolge eingelesen, wie sie in Tabelle 1 zu sehen sind. Aus Platzgründen werden hier nicht sämtliche verfügbare Zeichen des gesamten Fonts Latin Modern Math ausgegeben.

Tabelle 1: Tabellarische Anordnung der ersten 1360 Zeichen der LM Math

U+0020		!	"	#	\$	%	&	'	()
U+002a	*	+	,	-	.	/	0	1	2	3
U+0034	4	5	6	7	8	9	:	;	<	=
U+003e	>	?	@	A	B	C	D	E	F	G
U+0048	H	I	J	K	L	M	N	O	P	Q
U+0052	R	S	T	U	V	W	X	Y	Z	[
U+005c	\]	^	_	`	a	b	c	d	e
U+0066	f	g	h	i	j	k	l	m	n	o
U+0070	p	q	r	s	t	u	v	w	x	y
U+007a	z	{		}	~		i	e	£	¤
U+00a5	¥		§	¨	©	ª	«	¬	®	¯
U+00b0	°	±	´	µ	¶	·	¸	°	»	¼
U+00bd	½	¾	¿	À	Á	Â	Ã	Ä	Å	Æ
U+00c7	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï	Ð
U+00d1	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú
U+00db	Û	Ü	Ý	Þ	ß	à	á	â	ã	ä
U+00e5	å	æ	ç	è	é	ê	ë	ì	í	î
U+00ef	ï	ð	ñ	ò	ó	ô	õ	ö	÷	ø
U+00f9	ù	ú	û	ü	ý	þ	ÿ	Ā	ā	Ă
U+0103	ǻ	Ą	ą	Ć	ć	Č	č	Ď	ď	Đ
U+0112	Ē	ē	Ĕ	ė	Ė	ę	Ě	ě	Ĝ	ğ
U+0122	Ģ	ģ	Ĥ	ĥ	Ī	ī	Ĭ	ĭ	Į	į
U+0130	Ĳ	ı	Ĵ	ıj	Ķ	ķ	Ĺ	ĺ	Ļ	ļ
U+013d	Ľ	ł	Ł	ł	Ń	ń	Ņ	ņ	Ň	ň
U+014a	Đ	đ	Ō	ō	Ŏ	ö	Œ	œ	Ř	ř
U+0156	Ŕ	ŕ	Ŗ	ŗ	Ś	ś	Ş	ş	Š	š
U+0162	Ţ	ţ	Ť	ť	Ũ	ũ	Ū	ū	Ŭ	ŭ
U+0170	Ÿ	ÿ	Ų	ų	Ŷ	ŷ	Ž	ž	Ž	ž
U+017e	ž	ı	Ŏ	σ	Ū	ı	Ş	ş	Ţ	ţ
U+0237	Ĵ	ˆ	˘	˙	˚	˛	˜	˝	˞	˟
U+0301	˛	ˆ	˘	˙	˚	˛	˜	˝	˞	˟
U+030b	˝	˞	˟	ˠ	ˡ	ˢ	ˣ	ˤ	˥	˦
U+0332	˦	˧	/	=	A	B	Γ	Δ	E	Z

U+0397	H	Θ	I	K	Λ	M	N	Ξ	O	Π
U+03a1	P	Σ	T	Υ	Φ	X	Ψ	Ω	α	β
U+03b3	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ
U+03bd	ν	ξ	ο	π	ρ	ς	σ	τ	υ	φ
U+03c7	χ	ψ	ω	ϑ	φ	Ϙ	ρ	ε	Α	α
U+1ea2	Ă	ă	Á	á	À	à	Ã	ã	Ã	ã
U+1eac	Â	â	Á	á	À	à	Ã	ã	Ã	ã
U+1eb6	Ä	ä	E	e	Ë	ë	Ë	ë	É	é
U+1ec0	È	è	Ë	ë	Ë	ë	Ê	ê	Ì	ì
U+1eca	İ	i	Œ	œ	Ö	ö	Ó	ó	Ò	ò
U+1ed4	Õ	õ	Ö	ö	Ô	ô	Ó	ó	Ò	ò
U+1ede	Ö	ö	Õ	õ	Œ	œ	U	u	Û	û
U+1ee8	Ū	ú	Û	û	Ū	ú	Ū	û	U	u
U+1ef2	ÿ	ÿ	Y	y	ÿ	ÿ	ÿ	ÿ	-	-
U+2016		'	'	'	"	"	"	†	‡	•
U+2026	...	% ₀₀	% ₀₀₀	/	\	<	>	※	?	^
U+2044	/	{	}	%	©	€	™	™	™	™
U+20db	↔	↔	↔	↔	↔	↔	↔	↔
U+210d	H	h	ħ	ℋ	ℋ	ℋ	ℋ	ℋ	ℋ	ℋ
U+2118	℘	℘	℘	℘	℘	℘	℘	℘	℘	℘
U+2126	Ω	Ω	Ω	Ω	Ω	Ω	Ω	Ω	Ω	Ω
U+2136	⊔	⊔	⊔	⊔	⊔	⊔	⊔	⊔	⊔	⊔
U+2192	→	↓	↔	↔	↗	↗	↘	↘	↖	↖
U+219e	←	→	↔	↔	↖	↖	↗	↗	↘	↘
U+21b6	↶	↷	↻	↻	↶	↶	↷	↷	↶	↶
U+21c2	↓	↓	↔	↔	↔	↔	↔	↔	↔	↔
U+21cd	≠	≠	≠	≠	≠	≠	≠	≠	≠	≠
U+21db	⇒	⇒	⇒	⇒	⇒	⇒	⇒	⇒	⇒	⇒
U+2208	∈	∈	■	Π	Π	Σ	-	≠	+	/
U+2216	\	*	o	•	√	∞	∞	∠	∠	∠
U+2223		†		≠	∧	∨	∩	∪	∫	∫
U+2234	∴	∴	∴	∴	≈	≈	≈	≈	≈	≈
U+2247	≠	≈	≈	≠	≠	≠	≠	≠	≠	≠
U+2257	≠	≠	≠	≠	≠	≠	≠	≠	≠	≠
U+226a	≠	≠	≠	≠	≠	≠	≠	≠	≠	≠
U+2277	≠	≠	≠	≠	≠	≠	≠	≠	≠	≠
U+2283	∩	∩	∩	≠	≠	≠	≠	≠	∩	∩

U+2291	⌈	⌊	⌋	⌌	⊕	⊖	⊗	⊘	⊙	⊚
U+229b	⊕	⊖	⊗	⊘	⊙	⊚	⊛	⊜	⊝	⊞
U+22a9	⋈	⋉	⋊	⋋	⋌	⋍	⋎	⋏	⋐	⋑
U+22bb	√	∧	∨	∩	∪	◇	⋅	⋆	⋇	⋈
U+22c9	×	×	×	×	∩	∪	∩	∪	∩	∪
U+22d3	≡	≡	≡	≡	≡	≡	≡	≡	≡	≡
U+22e0	≠	≠	≠	≠	≠	≠	≠	≠	≠	≠
U+22ee	⋯	⋯	⋯	⋯	∅	>				
U+231c	⌈	⌊	⌋	⌌	()	<	>	⌈	⌊
U+239d	⌈	⌊	⌋	⌌						
U+23a7	⌈	⌊	⌋	⌌						
U+23dc	⌈	⌊	⌋	⌌	⌈	⌊	⌋	⌌	⌈	⌊
U+25aa	•	▲	△	▶	▽	♣	♠	♣	♠	○
U+25e6	◦	◯	♠	♥	◇	♣	♠	♣	♠	♣
U+266f	♯	⊘	⊙	√	⊕					<
U+27e9	>	()	⊙	⊕	⊗	⊕	⊖	⊕	≡
U+2a7e	≡	≡	≡	≡	≡	≡	≡	≡	≡	≡
U+2a96	≡	≡	≡	≡	≡	≡	≡	≡	≡	≡
U+2ac6	≡	≡	≡	≡	≡	≡	≡	≡	≡	≡
U+01d400	A	B	C	D	E	F	G	H	I	J
U+01d40a	K	L	M	N	O	P	Q	R	S	T
U+01d414	U	V	W	X	Y	Z	a	b	c	d
U+01d41e	e	f	g	h	i	j	k	l	m	n
U+01d428	o	p	q	r	s	t	u	v	w	x
U+01d432	y	z	A	B	C	D	E	F	G	H
U+01d43c	I	J	K	L	M	N	O	P	Q	R
U+01d446	S	T	U	V	W	X	Y	Z	a	b
U+01d450	c	d	e	f	g	i	j	k	l	m
U+01d45b	n	o	p	q	r	s	t	u	v	w
U+01d465	x	y	z	A	B	C	D	E	F	G
U+01d46f	H	I	J	K	L	M	N	O	P	Q
U+01d479	R	S	T	U	V	W	X	Y	Z	a
U+01d483	b	c	d	e	f	g	h	i	j	k
U+01d48d	l	m	n	o	p	q	r	s	t	u
U+01d497	v	w	x	y	z	A	C	D	G	J
U+01d4a6	K	N	O	P	Q	S	T	U	V	W
U+01d4b3	X	y	Z	A	B	C	D	E	F	G

U+01d4d7	\mathcal{H}	\mathcal{I}	\mathcal{J}	\mathcal{K}	\mathcal{L}	\mathcal{M}	\mathcal{N}	\mathcal{O}	\mathcal{P}	\mathcal{Q}
U+01d4e1	\mathcal{R}	\mathcal{S}	\mathcal{T}	\mathcal{U}	\mathcal{V}	\mathcal{W}	\mathcal{X}	\mathcal{Y}	\mathcal{Z}	\mathcal{A}
U+01d505	\mathfrak{B}	\mathfrak{D}	\mathfrak{E}	\mathfrak{F}	\mathfrak{G}	\mathfrak{J}	\mathfrak{K}	\mathfrak{L}	\mathfrak{M}	\mathfrak{N}
U+01d512	\mathfrak{O}	\mathfrak{P}	\mathfrak{Q}	\mathfrak{S}	\mathfrak{T}	\mathfrak{U}	\mathfrak{W}	\mathfrak{X}	\mathfrak{Y}	\mathfrak{Z}
U+01d51e	\mathfrak{a}	\mathfrak{b}	\mathfrak{c}	\mathfrak{d}	\mathfrak{e}	\mathfrak{f}	\mathfrak{g}	\mathfrak{h}	\mathfrak{i}	\mathfrak{j}
U+01d528	\mathfrak{k}	\mathfrak{l}	\mathfrak{m}	\mathfrak{n}	\mathfrak{o}	\mathfrak{p}	\mathfrak{q}	\mathfrak{r}	\mathfrak{s}	\mathfrak{t}
U+01d532	\mathfrak{u}	\mathfrak{v}	\mathfrak{w}	\mathfrak{x}	\mathfrak{y}	\mathfrak{z}	\mathfrak{A}	\mathfrak{B}	\mathfrak{D}	\mathfrak{E}
U+01d53d	\mathfrak{F}	\mathfrak{G}	\mathfrak{I}	\mathfrak{J}	\mathfrak{K}	\mathfrak{L}	\mathfrak{M}	\mathfrak{O}	\mathfrak{S}	\mathfrak{T}
U+01d54c	\mathfrak{U}	\mathfrak{V}	\mathfrak{W}	\mathfrak{X}	\mathfrak{Y}	\mathfrak{a}	\mathfrak{b}	\mathfrak{c}	\mathfrak{d}	\mathfrak{e}
U+01d557	\mathfrak{f}	\mathfrak{g}	\mathfrak{h}	\mathfrak{i}	\mathfrak{j}	\mathfrak{k}	\mathfrak{l}	\mathfrak{m}	\mathfrak{n}	\mathfrak{o}
U+01d561	\mathfrak{p}	\mathfrak{q}	\mathfrak{r}	\mathfrak{s}	\mathfrak{t}	\mathfrak{u}	\mathfrak{v}	\mathfrak{w}	\mathfrak{x}	\mathfrak{y}
U+01d56b	\mathfrak{z}	\mathfrak{A}	\mathfrak{B}	\mathfrak{C}	\mathfrak{D}	\mathfrak{E}	\mathfrak{F}	\mathfrak{G}	\mathfrak{H}	\mathfrak{I}
U+01d575	\mathfrak{J}	\mathfrak{K}	\mathfrak{L}	\mathfrak{M}	\mathfrak{N}	\mathfrak{O}	\mathfrak{P}	\mathfrak{Q}	\mathfrak{R}	\mathfrak{S}
U+01d57f	\mathfrak{T}	\mathfrak{U}	\mathfrak{W}	\mathfrak{X}	\mathfrak{Y}	\mathfrak{Z}	\mathfrak{a}	\mathfrak{b}	\mathfrak{c}	
U+01d589	\mathfrak{d}	\mathfrak{e}	\mathfrak{f}	\mathfrak{g}	\mathfrak{h}	\mathfrak{i}	\mathfrak{j}	\mathfrak{k}	\mathfrak{l}	\mathfrak{m}
U+01d593	\mathfrak{n}	\mathfrak{o}	\mathfrak{p}	\mathfrak{q}	\mathfrak{r}	\mathfrak{s}	\mathfrak{t}	\mathfrak{u}	\mathfrak{v}	\mathfrak{w}
U+01d59d	\mathfrak{x}	\mathfrak{y}	\mathfrak{z}	\mathfrak{A}	\mathfrak{B}	\mathfrak{C}	\mathfrak{D}	\mathfrak{E}	\mathfrak{F}	\mathfrak{G}
U+01d5a7	\mathfrak{H}	\mathfrak{I}	\mathfrak{J}	\mathfrak{K}	\mathfrak{L}	\mathfrak{M}	\mathfrak{N}	\mathfrak{O}	\mathfrak{P}	\mathfrak{Q}
U+01d5b1	\mathfrak{R}	\mathfrak{S}	\mathfrak{T}	\mathfrak{U}	\mathfrak{V}	\mathfrak{W}	\mathfrak{X}	\mathfrak{Y}	\mathfrak{Z}	\mathfrak{a}
U+01d5bb	\mathfrak{b}	\mathfrak{c}	\mathfrak{d}	\mathfrak{e}	\mathfrak{f}	\mathfrak{g}	\mathfrak{h}	\mathfrak{i}	\mathfrak{j}	\mathfrak{k}
U+01d5c5	\mathfrak{l}	\mathfrak{m}	\mathfrak{n}	\mathfrak{o}	\mathfrak{p}	\mathfrak{q}	\mathfrak{r}	\mathfrak{s}	\mathfrak{t}	\mathfrak{u}
U+01d5cf	\mathfrak{v}	\mathfrak{w}	\mathfrak{x}	\mathfrak{y}	\mathfrak{z}	\mathfrak{A}	\mathfrak{B}	\mathfrak{C}	\mathfrak{D}	\mathfrak{E}
U+01d5d9	\mathfrak{F}	\mathfrak{G}	\mathfrak{H}	\mathfrak{I}	\mathfrak{J}	\mathfrak{K}	\mathfrak{L}	\mathfrak{M}	\mathfrak{N}	\mathfrak{O}
U+01d5e3	\mathfrak{P}	\mathfrak{Q}	\mathfrak{R}	\mathfrak{S}	\mathfrak{T}	\mathfrak{U}	\mathfrak{V}	\mathfrak{W}	\mathfrak{X}	\mathfrak{Y}
U+01d5ed	\mathfrak{Z}	\mathfrak{a}	\mathfrak{b}	\mathfrak{c}	\mathfrak{d}	\mathfrak{e}	\mathfrak{f}	\mathfrak{g}	\mathfrak{h}	\mathfrak{i}
U+01d5f7	\mathfrak{j}	\mathfrak{k}	\mathfrak{l}	\mathfrak{m}	\mathfrak{n}	\mathfrak{o}	\mathfrak{p}	\mathfrak{q}	\mathfrak{r}	\mathfrak{s}
U+01d601	\mathfrak{t}	\mathfrak{u}	\mathfrak{v}	\mathfrak{w}	\mathfrak{x}	\mathfrak{y}	\mathfrak{z}	\mathfrak{A}	\mathfrak{B}	\mathfrak{C}
U+01d60b	\mathfrak{D}	\mathfrak{E}	\mathfrak{F}	\mathfrak{G}	\mathfrak{H}	\mathfrak{I}	\mathfrak{J}	\mathfrak{K}	\mathfrak{L}	\mathfrak{M}
U+01d615	\mathfrak{N}	\mathfrak{O}	\mathfrak{P}	\mathfrak{Q}	\mathfrak{R}	\mathfrak{S}	\mathfrak{T}	\mathfrak{U}	\mathfrak{V}	

Einbindung der Schrift

Sowohl für $\text{X}_{\text{L}}\text{A}_{\text{T}}\text{E}_{\text{X}}$ als auch $\text{L}_{\text{u}}\text{A}_{\text{T}}\text{E}_{\text{X}}$ empfiehlt sich das Laden des von Will Robertson und Philipp Stephani entwickelten Pakets `unicode-math`. Es vereinfacht die Schrifteinbindung erheblich:

```
\usepackage{fontspec}
\usepackage{unicode-math}
\setmathfont{Latin Modern Math}
```

Das Paket verfügt über eine variable Einbindung weiterer mathematischer Zeichensätze, falls die Hauptschrift nicht alle erforderlichen Zeichen enthält. unicode-math definiert die in der Tabelle 2 zusammengestellten Makros für eine Anwendung im Mathematikmodus, die nicht zwingend von jedem mathematischen Font unterstützt werden müssen (vergleiche den folgenden Abschnitt).

Tabelle 2: Zusammenfassung der mathematischen Schriftmakros und ihre Auswirkung am Beispiel $Aa\alpha\Gamma1$ für Latin Modern Math. In der Spalte *Alphabet* ist angegeben, für welche das Makro relevant ist: (L)atin, (G)reek und/oder (D)igits.

<i>Makro</i>	<i>Alphabet</i>	<i>Beispiel</i>
<code>\mathup</code>	LGD	$Aa\alpha\Gamma1$
<code>\mathbfup</code>	LGD	$Aa\alpha\Gamma1$
<code>\mathit</code>	LG	$Aa\alpha\Gamma1$
<code>\mathbfit</code>	LG	$Aa\alpha\Gamma1$
<code>\mathsfup</code>	LD	$Aa\alpha\Gamma1$
<code>\mathbfsfup</code>	LD	$Aa\alpha\Gamma1$
<code>\mathsfit</code>	L	$Aa\alpha\Gamma1$
<code>\mathbfsfit</code>	L	$Aa\alpha\Gamma1$
<code>\mathtt</code>	LD	$Aa\alpha\Gamma1$
<code>\mathbb</code>	LD	$Aa\alpha\Gamma1$
<code>\mathbbit</code>	L	$Aa\alpha\Gamma1$
<code>\mathscr</code>	L	$Aa\alpha\Gamma1$
<code>\mathbfscr</code>	L	$Aa\alpha\Gamma1$
<code>\mathfrak</code>	L	$Aa\alpha\Gamma1$
<code>\mathbffrak</code>	L	$Aa\alpha\Gamma1$

Fehlende Zeichen

Die Tabelle 2 zeigt, dass Latin Modern Math die Zeichen für Groß- und Kleinbuchstaben in Frakturschrift enthält, sowohl in der normalen als auch der fetten Variante.

`\mathfrak`: $\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$

`\mathfrak`: $\mathfrak{abcdefghijklmnopqrstuvwxyz}$

`\mathbffrak`: **$\mathfrak{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$**

`\mathbffrak`: **$\mathfrak{abcdefghijklmnopqrstuvwxyz}$**

Es fehlen jedoch die Kleinbuchstaben für die Skriptzeichen, welche durch die normalen Buchstaben ersetzt werden:

`\mathscr`: $\mathscr{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$

`\mathscr`: $\mathscr{abcdefghijklmnopqrstuvwxyz}$

`\mathbfscr`: **$\mathscr{ABCDEFGHIJKLMNOPQRSTUVWXYZ}$**

`\mathbfscr`: **$\mathscr{abcdefghijklmnopqrstuvwxyz}$**

Ist man auf sie angewiesen, so können die Skriptzeichen aus einem alternativen Mathematikfont geladen werden, indem *nach* der Definition von Latin Modern Math beispielsweise die Mathematikschrift Cambria Math für `\mathscr` definiert wird:

```
\usepackage{fontspec}
\usepackage{unicode-math}
\setmathfont{Latin Modern Math}
\setmathfont[range={\mathscr,\mathbfscr}]{Cambria Math}
```

Man erhält dann die gewünschten Skriptzeichen in normaler und fetter Form:

```
\mathscr: ABCDEFGHIJKLMNOPQRSTUVWXYZ
\mathscr: abcdefghijklmnopqrstuvwxyz
\mathbfscr: ABCDEFGHIJKLMNOPQRSTUVWXYZ
\mathbfscr: abcdefghijklmnopqrstuvwxyz
```

Die Schrift Cambria Math wiederum hat ebenso wie Latin Modern Math keine speziellen kalligrafischen Zeichen für `\mathcal`, die ohnehin in der Regel nicht Bestandteil einer mathematischen Schrift sind. Man findet sie allerdings in dem mathematischen Zeichensatz von XITS Math, wo es sie zudem in zwei Varianten gibt. Kleinbuchstaben sind hier allerdings nicht möglich.

```
\setmathfont[range={\mathcal,\mathbfcal},StylisticSet=1]{XITS Math}
```

```
StylisticSet=1: ABCDEFGHIJKLMNOPQRSTUVWXYZ
StylisticSet=1: ABCDEFGHIJKLMNOPQRSTUVWXYZ
StylisticSet=2: ABCDEFGHIJKLMNOPQRSTUVWXYZ
StylisticSet=2: ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

Literatur und Software

- [1] Roberto Ierusalimschy: *Programmieren mit Lua*; Open Source Press; München; 2006.
- [2] Roberto Ierusalimschy, Luiz Hernandez de Figueiredo und Waldemar Celes: *Reference Manual 5.1*; lua.org; Rio de Janeiro, Brasilien; 2006.
- [3] Will Robertson und Philipp Stephani: *Experimental Unicode mathematical typesetting: The unicode-math package*; CTAN:macros/latex/contrib/unicode-math/; Jan. 2011.
- [4] LuaTeX Development Team: *LuaTeX Reference*; CTAN:systems/luatex/base/manual/luatexref-t.pdf; Mai 2011.

Ausgabe vorhandener OpenType- oder TrueType-Schriften einer Schriftfamilie mit Lua^AT_EX

Herbert Voß

Je nach vorhandenem Betriebssystem kann die Zahl der vorhandenen Schriften sehr groß sein, so dass es manchmal schwer fällt, den Überblick zu behalten. Mit diesem Artikel wird gezeigt, wie man sich mit einer Lua-Funktion die vorhandenen Schriften einer Familie ausgeben lassen kann.

Die Liste aller Schriften einer Familie

Das Lua_TE_X-Paket `luaotfload`, das entsprechenden Code für das Laden von Systemschriften enthält und beispielsweise von `fontspec` benutzt wird, erstellt eine Liste `otfl-names.lua` aller dem System zur Verfügung stehenden Fonts. Dieses Schriftverzeichnis wird zum einen erstellt, wenn es noch nicht existiert oder zum anderen aktualisiert, wenn die angeforderte Schrift nicht in diesem Verzeichnis zu finden ist. Die Datei `otfl-names.lua` wird unter T_EX Live 2011 und Linux/Windows im Verzeichnis `$HOME/.texlive2011/texmf-var/luatex-cache/generic/names/` gespeichert. Bei Windows und Mik_TE_X findet man diese Liste in `<UserData>/luatex-cache/generic/names`. Die Liste enthält in dem Vektor `mappings` mehrere Angaben zu jeder Schrift, beispielsweise für `Delicious-Bold`:

```
{
  ["familyname"]="Delicious",
  ["filename"]={ "Delicious-Bold.otf", false },
  ["fontname"]="Delicious-Bold",
  ["fullname"]="Delicious-Bold",
  ["names"]={
    ["family"]="Delicious",
    ["fullname"]="Delicious-Bold",
    ["psname"]="Delicious-Bold",
    ["subfamily"]="Bold",
  },
  ["size"]={},
  ["slant"]=0,
  ["weight"]=700,
  ["width"]=5,
}
```

Diese Liste kann man innerhalb eines T_EX-Dokuments über einen Lua-Code einlesen und dann mit der gewünschten Schriftfamilie in einer `if`-Abfrage testen, ob ein Mustertext ausgegeben werden soll. Die Lua-Funktion wird über das

Makro `\directlua` aufgerufen, die dann über die Lua-Funktion `tex.print` eine Ausgabe innerhalb des T_EX-Dokuments generieren kann. Eingebettet wird der Lua-Code in die Umgebung `luacode`, die vom gleichnamigen Paket `luacode` zur Verfügung gestellt wird:

```
\begin{luacode}
local teststring = "Deutschsprachige Anwendervereinigung \TeX\ e.\,V."
local myfonts=dofile(fonts.names.path.localdir.."/otfl-names.lua")
function printFontDemo(fontname)
  local fnA = string.lower(fontname)
  local a = {}
  local b = {}
  for i, v in ipairs(myfonts.mappings) do
    m, n = string.find(string.lower(v.names.family), fnA)
    if m then
      a[v.fontname] = v.familyname
      b[#b + 1] = v.fontname
    end
  end
  table.sort(b)
  for i, v in ipairs(b) do
    tex.print("\parbox{0.2\linewidth}{\LMLL\footnotesize}")
    tex.print(-2, a[v])
    tex.print("|\scriptsize")
    tex.print(-2, v)
    tex.print("}\parbox{0.8\linewidth}{\raggedright\fontspec{" .. v .. "}")
    tex.print(teststring)
    tex.print("}\!|\!|/2pt}")
  end
end
\end{luacode}
```

Über einen Stringvergleich wird getestet, ob die zu testende Familie Teil der aktuellen Schleifenvariablen `v.names.family` ist. Ist dies der Fall, dann wird die Abfrage `if m` zu einer wahren Aussage und die entsprechenden Schriftinformationen werden in den beiden Tabellen `a` und `b` gespeichert, wobei `b` den Schriftnamen und `a` die Familie in Abhängigkeit des Schriftnamens speichert. Danach wird die Tabelle mit den Schriftnamen sortiert und dann anschließend mit der zugehörigen Schriftfamilie und der Testzeichenkette ausgegeben.

Anwendung

Insbesondere die Schriftfamilie Latin Modern kommt mit sehr vielen und die Schriftfamilie Biolinum mit ungewöhnlichen Varianten daher, die dem T_EX-Anwender nicht immer präsent sind. Mit dem Aufruf

```
\directlua{printFontDemo("Latin Modern")}
\directlua{printFontDemo("BioLi")}
\directlua{printFontDemo("Akko")}
```

kann jetzt jede Schrift testweise ausgegeben werden, die in ihrem Familiennamen *Latin Modern* beziehungsweise im zweiten Beispiel *Bioli* enthält, wobei nur darauf zu achten ist, dass die hier angegebene Zeichenkette möglichst eindeutig ist, auch wenn sie grundsätzlich nicht vollständig sein muss:

Alle Latin Modern Schriften

LM Math	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMath-Regular	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMono10	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMono10-Italic	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMono10-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMono12	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMono12-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMono8	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMono8-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMono9	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMono9-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoCaps10	DEUTSCHSPRACHIGE ANWENDERVEREINIGUNG T_EX E. V.
LMMonoCaps10-Oblique	<i>DEUTSCHSPRACHIGE ANWENDERVEREINIGUNG T_EX E. V.</i>
LMMonoCaps10	DEUTSCHSPRACHIGE ANWENDERVEREINIGUNG T_EX E. V.
LMMonoCaps10-Regular	DEUTSCHSPRACHIGE ANWENDERVEREINIGUNG T_EX E. V.
LMMonoLt10	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoLt10-Bold	Deutschsprachige Anwendervereinigung T_EX e. V.
LMMonoLt10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoLt10-BoldOblique	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoLt10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoLt10-Oblique	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoLt10	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoLt10-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoLtCond10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoLtCond10-Oblique	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoLtCond10	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoLtCond10-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoProp10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoProp10-Oblique	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoProp10	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoProp10-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoPropLt10	Deutschsprachige Anwendervereinigung T_EX e. V.
LMMonoPropLt10-Bold	Deutschsprachige Anwendervereinigung T_EX e. V.
LMMonoPropLt10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoPropLt10-BoldOblique	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoPropLt10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoPropLt10-Oblique	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoPropLt10	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoPropLt10-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMMonoSlant10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMMonoSlant10-Regular	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRoman10	Deutschsprachige Anwendervereinigung T_EX e. V.
LMRoman10-Bold	Deutschsprachige Anwendervereinigung T_EX e. V.
LMRoman10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRoman10-BoldItalic	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRoman10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRoman10-Italic	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>

LMRoman10	Deutschsprachige Anwendervereinigung T _E X e. V.
LMRoman10-Regular	
LMRoman12	Deutschsprachige Anwendervereinigung T_EX e. V.
LMRoman12-Bold	
LMRoman12-Italic	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRoman12-Regular	Deutschsprachige Anwendervereinigung T _E X e. V.
LMRoman17	Deutschsprachige Anwendervereinigung T _E X e. V.
LMRoman17-Regular	
LMRoman5	Deutschsprachige Anwendervereini-
LMRoman5-Bold	gung T_EX e. V.
LMRoman5	Deutschsprachige Anwendervereinigung
LMRoman5-Regular	T _E X e. V.
LMRoman6	Deutschsprachige Anwendervereinigung
LMRoman6-Bold	T_EX e. V.
LMRoman6	Deutschsprachige Anwendervereinigung T _E X
LMRoman6-Regular	e. V.
LMRoman7	Deutschsprachige Anwendervereinigung T_EX
LMRoman7-Bold	e. V.
LMRoman7-Italic	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRoman7	Deutschsprachige Anwendervereinigung T _E X e. V.
LMRoman7-Regular	
LMRoman8	Deutschsprachige Anwendervereinigung T_EX
LMRoman8-Bold	e. V.
LMRoman8	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRoman8-Italic	
LMRoman8	Deutschsprachige Anwendervereinigung T _E X e. V.
LMRoman8-Regular	
LMRoman9	Deutschsprachige Anwendervereinigung T_EX e. V.
LMRoman9-Bold	
LMRoman9	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRoman9-Italic	
LMRoman9	Deutschsprachige Anwendervereinigung T _E X e. V.
LMRoman9-Regular	
LMRomanCaps10	<i>DEUTSCHSPRACHIGE ANWENDERVEREINIGUNG T_EX E. V.</i>
LMRomanCaps10-Oblique	
LMRomanCaps10	DEUTSCHSPRACHIGE ANWENDERVEREINIGUNG T _E X E. V.
LMRomanCaps10-Regular	
LMRomanDemi10	Deutschsprachige Anwendervereinigung T_EX e. V.
LMRomanDemi10-Oblique	
LMRomanDemi10	Deutschsprachige Anwendervereinigung T_EX e. V.
LMRomanDemi10-Regular	
LMRomanDunh10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRomanDunh10-Oblique	
LMRomanDunh10	Deutschsprachige Anwendervereinigung T _E X e. V.
LMRomanDunh10-Regular	
LMRomanSlant10	Deutschsprachige Anwendervereinigung T_EX e. V.
LMRomanSlant10-Bold	
LMRomanSlant10	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LMRomanSlant10-Regular	

LMRomanSlant12	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanSlant12-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanSlant17	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanSlant17-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanSlant8	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanSlant8-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanSlant9	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanSlant9-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanUnsl10	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMRomanUnsl10-Regular	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSans10	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSans10-Bold	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSans10	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans10-BoldOblique	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans10	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans10-Oblique	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans10-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans12	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans12-Oblique	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans12	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans12-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans17	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans17-Oblique	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans17	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans17-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans8	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans8-Oblique	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans8	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans8-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans9	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans9-Oblique	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans9	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSans9-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSansDemiCond10	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSansDemiCond10-Oblique	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSansDemiCond10	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSansDemiCond10-Regular	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSansQuot8	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSansQuot8-Bold	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSansQuot8	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSansQuot8-BoldOblique	Deutschsprachige Anwendervereinigung \TeX e. V.
LMSansQuot8	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSansQuot8-Oblique	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSansQuot8	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMSansQuot8-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMTypewriter10 Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>
LMTypewriter10-Regular	<i>Deutschsprachige Anwendervereinigung \TeX e. V.</i>

Alle Biolinum Schriften

Linux Biolinum	Deutschsprachige Anwendervereinigung \TeX e. V.
LinBiolinum	

Linux Biolinum	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumB	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum	D e u t s c h s p r a c h i g e A n w e n -
LinBiolinumI	d e r v e r e i n i g u n g T_EX e . V .
Linux Biolinum Kb	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumKb	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum O	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumO	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum OB	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumOB	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum OI	D e u t s c h s p r a c h i g e A n w e n -
LinBiolinumOI	d e r v e r e i n i g u n g T_EX e . V .
Linux Biolinum Keyboard	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
O	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumOKb	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum Slanted	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumSl	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum Slanted O	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumSlantedO	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum Slanted OB	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumSlantedOB	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum Outline O	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumOO	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum Outline OOB	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumOOB	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum Outline OOI	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumOOI	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum Shadow O	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumWO	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
Linux Biolinum Shadow OI	Deutschsprachige Anwendervereinigung T_EX e. V.
LinBiolinumWOI	<i>Deutschsprachige Anwendervereinigung T_EX e. V.</i>
LinBiolinumWOB	
Linux Biolinum Shadow OOB	
LinBiolinumWOOB	
LinBiolinumWOI	

Alle Akko Pro Schriften

Akko Pro Light	Deutschsprachige Anwendervereinigung T _E X e. V.
AkkoPro-Light	
Akko Pro	Deutschsprachige Anwendervereinigung T _E X e. V.
AkkoPro-Regular	
Akko Rounded Pro Light	Deutschsprachige Anwendervereinigung T _E X e. V.
AkkoRoundedPro-Light	
Akko Rounded Pro	Deutschsprachige Anwendervereinigung T _E X e. V.
AkkoRoundedPro-Regular	

Schriftverzeichnisse

Bislang waren die Verzeichnisse, in denen T_EX nach Schriften gesucht hat, faktisch komplett von denen des Betriebssystems getrennt. Die alten T_EX-Engines stellen ganz spezifische Ansprüche an Schriften: Bevor eine Schrift genutzt werden kann, müssen die t_fm- (T_EX Font Metric), die v_f-Dateien (Virtual Font) die enc-

(Kodierungsvektoren) und die map-Datei (Zuordnung zu den externen Dateien) erstellt und vor allem im T_EX-Dateibaum installiert werden. Daher ist es sinnvoll, wenn auch die eigentlichen Schriftdateien im T_EX-Baum gespeichert und vom T_EX-System organisiert werden. Im Gegensatz dazu können X_YT_EX und LuaT_EX OpenType- und TrueType-Schriften direkt verwenden, ohne all diese Vorarbeiten und ohne das Erstellen gesonderter Dateien. Die Schriften sollten dann, wie bei anderen Programmen üblich, auch in normalen Systemordnern des aktuellen Betriebssystems gesucht werden.

Mit LuaT_EX ist die Lage etwas unübersichtlich, da im Skript von luaotfload Umgebungsvariablen, Betriebssystem und eventuell vorhandene Einträge in der Konfigurationsdatei `texmf.cnf` eine Rolle spielen können. Normalerweise werden die Schrift-Systemordner und die TrueType- und OpenType-Schriftordner in den T_EX-Bäumen durchsucht. Eine Konfigurationsmöglichkeit im eigentlichen Sinne gibt es nicht; in T_EX Live kann man immerhin noch weitere Suchpfade hinzufügen, indem man sie an die Variable `OSFONTDIR` in `texmf.cnf` anhängt. Einen genaueren Überblick über die Suchpfade von luaotfload bekommt man, wenn man auf der Kommandozeile die Datenbank einmal manuell erzeugt und sich dann die Logdatei anschaut.

```
mkluatexfontdb -v >lua-font-search.log
```

Ein manueller Aufruf von `mkluatexfontdb` ist auch dann nützlich, wenn luaotfload beim Erzeugen des Fontcaches hängen bleibt. Es lassen sich damit fehlerhafte Schriften identifizieren. Die problematische Schrift kann man dann anschließend in die Datei `otfl-blacklist.cnf` eintragen, um sie von der Suche auszuschließen. Man sollte sich immer bemühen, TrueType- und OpenType-Versionen einer Schrift nur in einem Verzeichnis, vorzugsweise in dem Systemverzeichnis, zu installieren.

Literatur und Software

- [1] Roberto Ierusalimsky: *Programmieren mit Lua*; Open Source Press; München; 2006.
- [2] Roberto Ierusalimsky, Luiz Hernandez de Figueiredo und Waldemar Celes: *Reference Manual 5.1*; lua.org; Rio de Janeiro, Brasilien; 2006.
- [3] LuaT_EX Development Team: *LuaT_EX Reference*; CTAN: systems/luatex/base/manual/luatexref-t.pdf; Mai 2011.

Tipps und Tricks

Rechtschreibprüfung mit T_EXworks

Herbert Voß

Mit der Installation von T_EX Live oder MiK_TE_X wird unter Windows auch automatisch der grafische Editor T_EXworks installiert. Für Linux hängt dies von der verwendeten Distribution ab, ob T_EXworks bereits installiert ist oder dies erst nachgeholt werden muss, beispielsweise für OpenSuSE durch `sudo zypper install texworks`. Jede Linux-Distribution hat hierbei ihre eigenen Programme und/oder Verfahren zur Installation von Programmen.

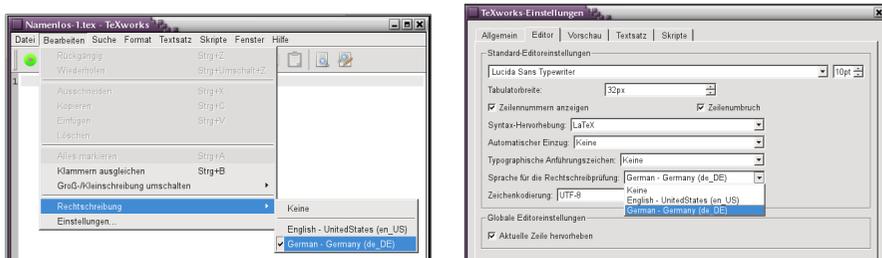
T_EXworks wird bei T_EX Live in der Regel ohne die notwendigen Lexika für eine Rechtschreibkontrolle installiert. Verwendet werden können die mit dem ehemaligen OpenOffice ausgelieferten Lexika, die jetzt unter <http://extensions.services.openoffice.org/en/dictionaries> verfügbar sind. Möglich sind auch die mit dem Programm thunderbird mitgelieferten Dateien mit den Endungen *.aff und *.dic, die sich für Linux in einem Unterverzeichnis `$HOME/.thunderbird/.../extensions/.../` befinden. Bei einer Standardinstallation von T_EX Live liegen die Konfigurationsdateien für T_EXworks unter `$HOME/.texlive2011/texmf-config/texworks/`. Hierin befindet sich ein Unterverzeichnis `dictionaries/`, in welches die Lexika zu speichern sind. Die entsprechenden Verzeichnisse kann man sich auch durch den Menüpunkt Hilfe▷Einstellungen und Ressourcen anzeigen lassen.

Für Linux kommen die Dateien nach `$HOME/.Texworks/dictionaries/`. Damit sollten dann die Sprachen für T_EXworks zur Verfügung stehen, wobei eventuell ein Neustart nötig ist, falls das Programm bereits vor Installation der Lexika aktiv war.

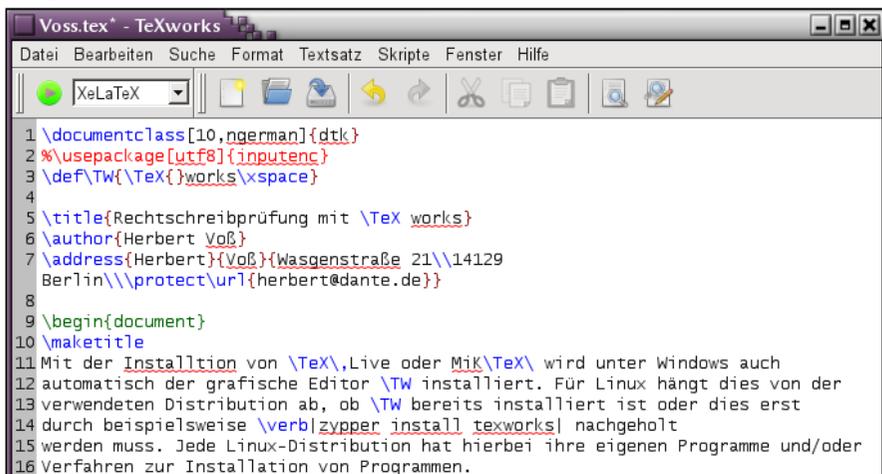
Bei einer MiK_TE_X-Installation für Windows werden mit der aktuellen Version 2.9 bereits Lexika für Englisch (Amerikanisch), Französisch und Deutsch mitgeliefert, die aber normalerweise im Verzeichnis `C:\Programme\MiKTeX2.9\hunspell\` liegen. Zusätzliche Lexika lassen sich dann alternativ dort speichern oder in dem entsprechenden Verzeichnis von T_EXworks, welches man bei einer Standard-

installation von MiK \TeX unter C:\Users\\TeX\2.9\TeXworks\0.4\ dictionaries findet.

In \TeX works kann eine Sprache nach dem Starten zum einen unter dem Menüpunkt Bearbeiten>Rechtschreibung für das aktuelle Dokument und zum anderen unter Bearbeiten>Einstellungen>Editor für alle Dokumente ausgewählt werden:



Die Rechtschreibüberprüfung erfolgt dann direkt nach der Eingabe des Textes, wobei unbekannte Wörter mit einer roten Wellenlinie unterlegt werden. Die Makros, die mit einem Backslash beginnen, werden dabei von der Rechtschreibprüfung ausgenommen, nicht jedoch eventuelle Argumente.



Literatur und Software

- [1] Jonathan Kew und Stefan Löffler: *TeXworks – A simple interface for working with TeX documents*; <http://code.google.com/p/texworks/>; Juli 2011.
- [2] Jonathan Kew und Stefan Löffler: *TeXworks – lowering the entry barrier to the TeX world*; <http://tug.org/texworks/>; Juli 2011.

Zerlegen einer Grafik in gleiche Teile

Herbert Voß

Eine gegebene Grafik lässt sich mit den Möglichkeiten des Paketes `graphicx` vielfach manipulieren, unter anderem auch in gleiche Teile zerlegen. Dies soll mit der folgenden Grafik gezeigt werden, die aus reinen Platzgründen hier auf die Hälfte verkleinert wurde.



Zuerst benötigt man die Maße der Originalgrafik, die dazu in eine Box gesetzt wird, sodass sich Breite und Höhe leicht bestimmen lassen:

```
\usepackage{graphicx}
\newsavebox\IBox
\savebox\IBox{\includegraphics{dante22}}
\newlength\Iwidth
\newlength\Iheight
\setlength\Iwidth{\wd\IBox}
\setlength\Iheight{\ht\IBox}
```

Diese Breite und Höhe der Grafik kann nun in beliebig viele ganzzahlige Teile zerlegt werden, wozu das Makro `\multido` aus dem gleichnamigen Paket verwendet wird. Hier wird willkürlich eine Zerlegung in eine 4×2 -Matrix gewählt. Die Angaben für den `viewport` müssen in Klammern gesetzt werden, da das folgende Leerzeichen sonst von \LaTeX entfernt wird und es somit zu einem Fehler käme. Die Skalierung von `scale=0.5` wird hier nur gewählt, da auch die Originalgrafik auf der vorhergehenden Seite entsprechend skaliert wurde. Für das Zerlegen der Grafik wird hingegen immer die Originalgröße benutzt.

```
\multido{\ry=0.5+-0.5, \rY=1.0+-0.5}{2}{%
```

```

\multido{\rx=0.00+0.25, \rX=0.25+0.25}{4}{%
  \fbox{\includegraphics[%
    viewport={\rx\width} {\ry\height} {\rX\width} \rY\height,
    clip,scale=0.5]{dante22}}
}

```



Zu beachten ist noch, dass in der erstellten PDF-Datei die Originalgrafik formal achtmal eingefügt wurde, denn nur durch die Option `clip` wird der überstehende Rand der Originalgrafik abgeschnitten. Dies lässt sich leicht überprüfen, wenn die Option entfernt wird. Weiterhin ist zu beachten, dass der \LaTeX -Treiber für `graphicx` die Option `clip` nicht unterstützt und \LaTeX somit in diesem Fall nicht verwendet werden kann.

Von fremden Bühnen

Neue Pakete auf CTAN

Jürgen Fenn

Der Beitrag stellt neue Pakete auf CTAN seit der letzten Ausgabe bis zum Redaktionsschluss vor. Die Updates können auf der *ctan-ann*-Mailingliste verfolgt werden, die auch über Twitter und Identi.ca als @ctanannounce verfügbar sind.

fontbook von *Raphaël Pinson* erzeugt eine Übersicht über die mit X_YTeX nutzbaren OpenType-Fonts.

CTAN:macros/xetex/latex/fontbook

sapthesis von *Francesco Biccari* ist die L^AT_EX-Klasse für Abschlussarbeiten der Universität Rom Sapienza.

CTAN:macros/latex/contrib/sapthesis

vak von *Vitaly G. Melnikov* ist ein BibT_EX-Stil für akademische Arbeiten in russischer Sprache nach den Standards GOST 7.82 und GOST 7.1.

CTAN:biblio/bibtex/contrib/vak

beamersubframe von *Mike Kaufmann* dient zum Ändern der Anordnung von Frames in einer PDF-Datei ohne Umstellen der Quelldatei einer Beamer-Präsentation.

CTAN:macros/latex/contrib/beamer-contrib/beamersubframe

chemnum von *Clemens Niederberger* dient zum Nummerieren von chemischen Gleichungen.

CTAN:macros/latex/contrib/chemnum

luabibentry von *Oliver Kopp* erlaubt das Wiederholen bibliographischer Angaben, ähnlich dem L^AT_EX-Paket *bibentry*.

CTAN:macros/luatex/latex/luabibentry

getoptk von *Michael Le Barbier Grünewald* ist ein Plain-T_EX-Paket, das ein Interface bereitstellt, mit dem Makros wie `\hbox`, `\hrule` definiert werden können: Befehle mit Optionen, die ihrerseits Argumente annehmen.

CTAN:macros/plain/contrib/getoptk

othelloboard von *Steven Hall* ist ein Paket zum Dokumentieren von Spielstellungen bei Othello/Reversi.

CTAN:macros/latex/contrib/othelloboard

tsemlines von Tobias Schlemmer hilft beim Setzen von \TeX cad-Dokumenten, die em \TeX -specials enthalten, welche von pdf \TeX nicht unterstützt werden. Derzeit wird `\emLines` emuliert.

CTAN:macros/latex/contrib/tsemlines

lshort-czech von Pavel Štříž ist die neue tschechische Übersetzung von *l2kurz*.

CTAN:info/lshort/czech

polyglossia-kannada von Aravinda ist eine Unterstützung für die Sprache Kannada für das \LaTeX -Paket *polyglossia*.

CTAN:macros/xetex/latex/polyglossia-contrib/kannada

serbian-date-lat von Zoran Filipovic gibt das Datum auf Serbisch aus, in lateinischer Schrift und in neuer Rechtschreibung.

CTAN:macros/latex/contrib/babel-contrib/serbian-date-lat

serbian-apostrophe von Zoran Filipovic stellt mehrere Befehle zur Eingabe serbischer Wörter bereit, die Akzente enthalten. Die Ausgabe erfolgt in lateinischer Schrift.

CTAN:language/serbian/serbian-apostrophe

dutchcal von Michael Sharpe ist eine Überarbeitung des kalligraphischen Mathematikfonts ESSTIX13, der um einen fetten Schnitt ergänzt wird, einschließlich der \LaTeX -Unterstützung.

CTAN:fonts/dutchcal

cascadilla von Max Bane ist die Klasse für Veröffentlichungen des *Cascadilla Proceedings Project* (Sprachwissenschaft).

CTAN:macros/latex/contrib/cascadilla

decorule von Peter Flynn dient zum Zeichnen von Zierzeilen, wie sie in Büchern im 19. Jahrhundert gebräuchlich waren (vgl. TUGboat 31 (1), 21–22).

CTAN:macros/latex/contrib/decorule

srbook-mem von Zoran Filipovic stellt Gliederungsbefehle in serbischer Sprache für `memoir` bereit.

CTAN:macros/latex/contrib/srbook-mem

pst-rubans von Manuel Luque und Herbert Voß dient zum dreidimensionalen Zeichnen von Bändern mit Hilfe von PSTricks.

CTAN:graphics/pstricks/contrib/pst-rubans

serbianc von Filip Brcic ist das erste »*babel contributed package*« auf CTAN: Eine Unterstützung für kyrillische Texte auf Serbisch.

CTAN:macros/latex/contrib/babel-contrib/serbianc

schwalbe-chess von Stefan Hönig dient zum Setzen der Zeitschrift der *Deutschen Vereinigung für Problemschach* »Die Schwalbe«.

CTAN:macros/latex/contrib/schwalbe-chess

ifnnextok von Uwe Lück ersetzt das interne \LaTeX -Makro `\@ifnnextchar` so, dass der Befehl `\ [` eine öffnende eckige Klammer in die neue Zeile aus gibt.

CTAN:macros/latex/contrib/ifnnextok

pst-layout von Michael Sharpe bietet Makros zur Gestaltung der »quasi-tabellarischen Drucksachen wie Visitenkarten, Speisekarten, Musikprogrammen, Broschüren« mit

Hilfe von PSTricks.

CTAN:graphics/pstricks/contrib/pst-layout

serbian-lig von *Zoran Filipovic* deaktiviert Ligaturen für einige serbische Begriffe bei Verwendung von lateinischen Schriftarten.

CTAN:macros/latex/contrib/serbian-lig

chet von *Andreas Stergiou* »vereinfacht das T_EX« und erzeugt das Layout der harv-mac-Makros.

CTAN:macros/latex/contrib/chet

upca von *Jiří Mrkus* enthält Makros zum Drucken von UPC-A-Strichcodes.

CTAN:macros/generic/upca

regstats von *Hans-Martin Münch* erlaubt es, die Zahl der verwendeten Register (Zähler etc.) auszugeben und mit der Zahl der insgesamt verfügbaren Register zu vergleichen. Es ähnelt dem Paket regcount.

CTAN:macros/latex/contrib/regstats

chemmacros von *Clemens Niederberger* ist eine Sammlung von Makros zum Setzen von chemischen Formeln.

CTAN:macros/latex/contrib/chemmacros

thumbs von *Hans-Martin Münch* dient zum Erstellen eines oder mehrerer Daumenindizes (*thumb index*).

CTAN:macros/latex/contrib/thumbs

xhfill von *Herbert Voß* erweitert den Befehl `\hrulefill` so, dass Breite und Farbe der dabei erzeugten Linie geändert werden können.

CTAN:macros/latex/contrib/xhfill

unamthesis von *Julio A. Freyre-Gonzalez* ist die Vorlage für Abschlussarbeiten der *Universidad Nacional Autónoma de México*.

CTAN:macros/latex/contrib/unamthesis

mathalfa von *Michael Sharpe* ist ein allgemeines Paket zum Aufruf von Mathematikschriften.

CTAN:macros/latex/contrib/mathalfa

grafcet von *Robert Papanicola* dient zum Zeichnen von GRAFCET-Diagrammen mit `pgf/TikZ`.

CTAN:graphics/pgf/contrib/grafcet

fdsymbol von *Michael Ummels* ist ein Symbolfont für den Mathematikmodus, der die Schrift Fedra von Typotheque ergänzt.

CTAN:fonts/fdsymbol

cantarell von *Mohamed El Morabity* ist die L^AT_EX-Unterstützung für die Schriftart Cantarell.

CTAN:fonts/cantarell

ant-worker-tasks von *Michael Niedermair* sind Tasks für *Apache-Ant*, um insbesondere PDF-Dateien zu bearbeiten.

CTAN:support/ant-worker/tasks

latex4wp-it von *Guido Gonzato* ist die italienische Übersetzung der Anleitung » \LaTeX for Word Processor Users«.

CTAN:info/latex4wp-it

esstix von *Michael Sharpe* enthält den gleichnamigen Font, einen Vorläufer der STIX Fonts.

CTAN:fonts/esstix

lualatex-math von *Philipp Stephani* enthält einige Berichtigungen für den mathematischen Formelsatz, die aus dem Paket `unicode-math` übernommen worden sind und die einige Probleme unter \LaTeX beseitigen.

CTAN:macros/luatex/latex/lualatex-math

boondox von *Michael Sharp* enthält eine aus den STIX-Fonts abgeleitete Schrift für den Mathematiksatz.

CTAN:fonts/boondox

lualatex-platform von *Philipp Stephani* ist ein Erweiterungsmodul für plattformspezifischen Code.

CTAN:systems/luatex/contrib/lualatex-platform

contribnuc von *Neal Davis* dient zum automatischen Setzen von Radioisotopen.

CTAN:macros/latex/contribnuc

uafthesis von *Joshua Holbrook* ist eine \LaTeX -Dokumentenklasse für Abschlussarbeiten der *University of Alaska Fairbanks*.

CTAN:macros/latex/contrib/uafthesis

uni-wtal-ger von *Carsten Ace Dahlmann* ist ein neuer `biblatex`-Stil, der auf `author-title-dw` beruht und für die literaturwissenschaftliche Fakultät der Bergischen Universität Wuppertal entwickelt wurde.

CTAN:macros/latex/exptl/biblatex-contrib/uni-wtal-ger

sitem von *Piotr Majkrzak* speichert den Inhalt eines `\item` in einer Box.

CTAN:macros/latex/contrib/sitem

cmpj von *Andrij Shvoika* dient zum Setzen von Beiträgen für die Zeitschrift *Condensed Matter Physics*.

CTAN:macros/latex/contrib/cmpj

Bücher und Rezensionen

»Schnell ans Ziel mit $\LaTeX 2\epsilon$ « von Jörg Knappen

Herbert Voß

Bei einer dritten Auflage eines Buches kann man sich eigentlich eine Rezension sparen, denn der Leser hat bereits sein Urteil abgegeben; die zweite Auflage war vergriffen. Insofern ist es etwas müßig, sich trotzdem auf die Seite des Kritikers zu begeben und auch entsprechend zu kritisieren.

Der Markt der Bücher, die als Einführung in das Satzsystem \TeX beziehungsweise \LaTeX gedacht sind, ist sehr übersichtlich; die Bücher von Niedermayr und Detig sind nicht mehr lieferbar, sodass nur noch wenige gute Bücher den Bereich der Einführung abdecken.

Die dritte Auflage des Buches von Jörg Knappen, die bereits im Jahr 2009 erschien, ist zwar eine Überarbeitung der zweiten, folgt aber mehr oder weniger dem Inhalt der alten Auflage. Die \TeX -Welt befindet sich voraussichtlich in einem kleinen bis mittelgroßen Umbruch. Mit \XeTeX und \LuaTeX stehen zwei Varianten bereit, die insbesondere in mehrsprachigen Systemen Vorteile bieten. 2009 war sicherlich noch nicht abzusehen, wie sich diese beiden Systeme entwickeln werden. Deshalb findet man auch nur eine Seite zu diesen beiden Neuentwicklungen von \TeX , obwohl man heute weiß, dass es hätten mehr sein können.

Das einführende Beispiel des Buches nimmt wenig Rücksicht auf Windows-Nutzer und verlangt Kenntnisse beim Umgang mit der Konsole, was heute nicht unbedingt vorausgesetzt werden kann. Ob man die Beispiele in der heutigen Zeit noch ohne eine entsprechende Eingabekodierung angeben sollte, erscheint mir doch sehr zweifelhaft, denn Texte lesen sich nun einmal flüssiger, wenn man



grundsätzlich »Hallo schöne große Welt« im Beispiel liest. Ein Vorteil des Buches ist sicherlich die Tatsache, dass der Schwierigkeitsgrad am Anfang sehr niedrig gehalten wird, sodass Frustrationen bei echten Anfängern wohl auszuschließen sind.

In dem Buch wird der Bereich Schriften verständlicherweise etwas ausführlicher gehalten, denn Jörg Knappen ist nun einmal hier »zu Hause«. Dass bei 270 Seiten die Ausführlichkeit bei allen Themen ihre Grenzen hat, versteht sich von selbst. Sollte es eine weitere Auflage geben, so ist dringend anzuraten, dass man den neueren Entwicklungen bei den Paketen, der Bibliografie und den grafischen Umgebungen breiteren Raum einräumt. Ob man wirklich mit dem Buch »schnell ans Ziel kommt«, muss letztlich jeder einzelne Leser selbst beurteilen. Das Buch ist bei 270 Seiten allerdings nicht gerade kostengünstig zu nennen.

Jörg Knappen
Schnell ans Ziel mit L^AT_EX₂e
3., überarbeitete Auflage
Oldenbourg Wissenschaftsverlag, München, 2009
270 Seiten
ISBN 978-3-486-59015-9; 29,80 €

Leserbriefe

Kyrillische Schriftzeichen im L^AT_EX-Dokument

Arno Trautmann

Sehr interessiert habe ich den informativen Artikel »Kyrillische Schriftzeichen im L^AT_EX-Dokument« in der DTK 1/2011 gelesen. Das Interesse war allerdings eher historisch. Die beschriebenen Methoden zum Einfügen kyrillischer Zeichen funktionieren zwar, sind aber umständlich und veraltet. Die Problematik zeigt sich schon bei den Abbildungen, in denen das Zeichen Т dargestellt wird; in der Beschriftung jedoch steht »Das kyrillische Zeichen đ«, und im Fließtext wird wiederum Т ausgegeben. Im Editorial wird auch angemerkt, der Artikel sei »eine Herausforderung für das Setzen« gewesen. Mit einem System wie L^AT_EX sollten sehr erfahrene und fähige Nutzer, wie die DTK-Redakteure, doch in der Lage sein, kyrillische Zeichen problemlos setzen zu können – da läuft doch etwas falsch? Der Grund liegt in veralteten, umständlichen Eingabe- und Schriftkodierungen, wie auch Fußnote 2 des Artikels zeigt: Die empfohlenen unterschiedlichen Kodierungen für verschiedene Betriebssysteme verhindern die Portabilität von Dokumenten. Warum wird nicht für jedes System utf8 empfohlen, was überall verfügbar ist? Auch das Verwenden von Schriftkodierungen mit jeweils nur 256 Zeichen beschränkt die Zeichenvielfalt und ist fehleranfällig.

Verwendet man LuaL^AT_EX oder XeL^AT_EX mit utf8-Eingabe und OpenType-Schriften, stellt sich die Problematik gar nicht: Jedes Zeichen kann, sofern die Schrift eine Glyphie anbietet, direkt eingegeben werden. Um das Zeichen Т zu setzen, ist lediglich folgender Code nötig:

```
\usepackage{fontspec}\setmainfont{Linux Libertine}
\begin{document}
Ein Dokument mit \char"0402.
\end{document}
```

Die Eingabe von Zeichen hat Dominik Wagenführ in seinem Leserbrief in der gleichen DTK-Ausgabe erläutert; man kann aber auch \char verwenden und den Codepunkt im Internet z. B. auf www.decodeunicode.org nachschauen. Das Zeichen steht dann normal im Text und es kann nicht passieren, dass Т zu đ wird.

So bietet sich eine einfache, stabile Methode, beliebige Zeichen in Dokumente beliebiger Sprachen einzufügen, ohne Kodierungen zu wechseln und zu riskieren, das ganze Dokument in falscher Kodierung zu setzen.

Dieser Brief ist auch eine Anregung an die DTK-Redaktion, den Umstieg auf lua \TeX anzudenken; entsprechende OpenType-Schriften werden ja bereits verwendet.

Anmerkung der Redaktion

Zu den verschiedenen Ausgabeformen der Zeichen ist anzumerken, dass in dem Artikel zum einen die lateinische und zum anderen die kyrillische Ausführung des Zeichens ausgegeben wurden, ohne dass darauf explizit verwiesen wurde. Es sind also keine falschen Zeichen in dem Artikel vorhanden.

Tabellen berechnen mit Lua \LaTeX

Roland Geiger

Die Vorschau auf Heft 2/2011 sagte: »Tabellen berechnen mit LuaLaTeX«. Entsprechend groß war die Erwartung auf das neue Heft, denn das Thema interessierte mich und so war ich gespannt auf den Artikel. Der Inhalt übertraf dann weit das, was ich mir vorgestellt hatte.

Nach dem ersten Durchlesen war klar, ich hatte nicht alles verstanden. Deshalb baute ich `rechnung.lua` und `rechnung.tex` 1:1 nach. Anschließend noch die Steuerdatei `rechnung.lco` für den Brief und ein Logo. Noch den zitierten Artikel DTK 2/2003 gelesen und die Steuerdatei für »Moderne Briefe« eingebunden, das Ergebnis sah genauso aus, wie abgebildet. Nun, ich hatte auch noch die zitierte Literatur zu sichten, im WWW nach der Formatierung von Lua-Strings zu suchen. Was bleibt: Den Lua-Code habe ich verstanden, auch wenn ich ihn nicht hätte schreiben können. Am Versuch, ihn zu debuggen (wie in bspw. VBA schrittweise ablaufen zu lassen und die Werte zu kontrollieren), bin ich gescheitert. Den Fehler im Lua-Script habe ich trotzdem gefunden.

In der Lua-Tabelle fehlt in Zeile 2 `Kopfdaten={}` und Zeile 3 `Positionen={}` jeweils ein Komma am Zeilenende.

Nun zum Schluss: Ein Dankeschön an die Redaktion für den Artikel und ein ganz besonderer Dank an die beiden Autoren.

Spielplan

Termine

2011

- 19. 9. – 24. 9. **5th International ConTeXt meeting**
Bassenge/Belgien
<http://meeting.contextgarden.net/2011/>
- 1. 10. **Herbsttagung**
und 45. Mitgliederversammlung von DANTE e.V.
Garmisch-Partenkirchen
<http://www.dante.de/events/mv45.html>
- 19. 10. – 21. 10. **TUG 2011**
Trivandrum/Indien (Kerala)
<http://tug.org/tug2011/>

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://projekte.dante.de/Stammtische/WebHome>.

Aachen

Torsten Bronger,
bronger@physik.rwth-aachen.de
Gaststätte Knossos, Templergraben 28, 52062 Aachen
Zweiter Donnerstag im Monat, 19.00 Uhr

Berlin

Rolf Niepraschk, Tel.: (030) 3 48 13 16,
Rolf.Niepraschk@gmx.de
Ort derzeit ungeklärt
Zweiter Donnerstag im Monat, 19.00 Uhr

Bremen

Winfried Neugebauer, Tel.: 0176 60 85 43 05,
tex@pfn.de
Wechselnder Ort
Erster Donnerstag im Monat, 18.30 Uhr

Darmstadt

Karlheinz Geyer, geyerk.fv.tu@nds.tu-darmstadt.de, <http://www.da-tex.org>
Wechselnder Ort
Erster Freitag im Monat, ab 19.30 Uhr

Erlangen

Walter Schmidt, Peter Seitz,
w.a.schmidt@gmx.net
Gaststätte »Deutsches Haus«, Luitpoldstraße 25,
91052 Erlangen
Dritter Dienstag im Monat, 19.00 Uhr

Hamburg

Lothar Fröhling, lothar@thefroehlings.de
Restaurant Sandstuw, Neue Straße 17, 21073
Hamburg-Harburg
Letzter Dienstag im Monat, 19.00 Uhr

Hannover

Mark Heisterkamp,
heisterkamp@rrzn.uni-hannover.de
Seminarraum RRZN, Schloßwender Straße 5, 30159
Hannover
Zweiter Donnerstag im Monat, 18.30 Uhr

Heidelberg

Martin Wilhelm Leidig, Tel.: (06203) 40 22 03,
moss@moss.in-berlin.de
Anmeldeseite zur Mailingliste: <http://mailman.moss.in-berlin.de/mailman/listinfo/stammtisch-hd-moss.in-berlin.de>
Wechselnder Ort
Letzter Freitag im Monat, ab 19.30 Uhr

Karlsruhe

Klaus Braune, Tel.: (0721) 608-4 40 31,
klaus.braune@kit.edu,
SCC (Steinbuch Centre for Computing) des KIT
(vormals Universität Karlsruhe, Rechenzentrum),
Zirkel 2, 2. OG, Raum 203, 76131 Karlsruhe
Erster Donnerstag im Monat, 19.30 Uhr

Köln

Helmuth Siegert
Institut für Kristallographie, Zülpicher Straße 49b,
50674 Köln
Letzter Dienstag im Monat, 19.30 Uhr

Konstanz

Kurt Lidwin, kurt.lidwin@web.de,
Restaurant Radieschen, Hohenhausgasse 1, 78462
Konstanz
Zweiter Dienstag im Monat, 19.00 Uhr

München

Uwe Siart, uwe.siart@tum.de, <http://www.siart.de/typografie/stammtisch.xhtml>
Erste Woche des Monats an wechselnden Tagen,
19.00 Uhr

Stuttgart

Bernd Raichle, bernd.raichle@gmx.de
Bar e Ristorante »Valle«, Geschwister-Scholl-Straße 3,
70197 Stuttgart
Zweiter Dienstag im Monat, 19.30 Uhr

Trier

Martin Sievers,
stammtisch-trier@texberatung.de
Fetzenkneipe (Haus Fetzenreich), Sichelstraße 36
(beim Sieh-Um-Dich), 54290 Trier
Dritter Montag des Monats, 20.15 Uhr

Wuppertal

Andreas Schrell, Tel.: (02193) 53 10 93,
as@schrell.de
Restaurant Croatia »Haus Johannisberg«, Südstraße
10, 42103 Wuppertal
Zweiter Donnerstag im Monat, 19.30 Uhr

Würzburg

Bastian Hepp, LaTeX@sning.de
nach Vereinbarung

Adressen

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

Tel.: (0 62 21) 2 97 66 (Mo., Mi.–Fr., 10.00–12.00 Uhr)

Fax: (0 62 21) 16 79 06

E-Mail: dante@dante.de

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

Präsidium

Präsident:	Volker RW Schaa	president@dante.de
Vizepräsident:	Adelheid Grob	vice-president@dante.de
Schatzmeister:	Klaus Höppner	treasurer@dante.de
Schriftführer:	Manfred Lotz	secretary@dante.de
Beisitzer:	Bernd Raichle	
	Martin Sievers	
	Herbert Voß	
	Uwe Ziegenhagen	

Server

CTAN: <http://mirror.ctan.org/>

DANTE: <http://www.dante.de/>

FAQ

DTK: <http://projekte.dante.de/DTK/WebHome>

T_EX: <http://projekte.dante.de/DanteFAQ/WebHome>

Autoren/Organisatoren

- Jürgen Fenn**
Friedensallee 174/20
63263 Neu-Isenburg
juergen.fenn@gmx.de
- Enrico Gregorio**
Dipartimento di Informatica
Università di Verona
enrico.gregorio@univr.it
- Adelheid Grob**
siehe Seite 82
- Andreas Hirsch**
Werdenfels-Gymnasium
A112
Wettersteinstraße 30
82467 Garmisch-Partenkirchen
mitarbeiter2@verwaltung.werdenfels-
gymnasium.de
- Axel Kielhorn**
Lesumstraße 10
27283 Verden
A.Kielhorn@web.de
- [72] **Reinhard Kotucha** [45]
Marschnerstr. 25
30167 Hannover
reinhard.kotucha@web.de
- [32] **Rolf Niepraschk** [49]
Persiusstr. 12
10245 Berlin
Rolf.Niepraschk@gmx.de
- [6] **Volker RW Schaa** [6, 7]
siehe Seite 82
- Herbert Voß** [3, 52, 60, 68, 70, 76]
Wasgenstraße 21
14129 Berlin
herbert@dante.de
- [20] **Dominik Wagenführ** [10]
dwagenfuehr@freiesmagazin.de

Die T_EXnische Komödie

23. Jahrgang Heft 3/2011 August 2011

Impressum

Editorial

Hinter der Bühne

- 6 Grußwort
- 8 Herbsttagung

Bretter, die die Welt bedeuten

- 10 Variable Argumente in L^AT_EX nutzen
- 21 Viele Ziele – *Multi Target Publishing*
- 33 Installation der T_EX Live 2011 auf Ubuntu
- 46 Installation nicht ganz freier Fonts
- 49 Trennmuster
- 52 Latin Modern Math
- 61 LuaL^AT_EX und Schriften

Tipps und Tricks

- 68 Rechtschreibprüfung mit T_EXworks
- 70 Zerlegen einer Grafik in gleiche Teile

Von fremden Bühnen

- 72 Neue Pakete auf CTAN

Bücher und Rezensionen

- 76 »Schnell ans Ziel mit L^AT_EX2e« von Jörg Knappen

Leserbriefe

- 78 Kyrillische Schriftzeichen im L^AT_EX-Dokument
- 79 Tabellen berechnen mit LuaL^AT_EX

Spielplan

- 80 Termine
- 81 Stammtische

Adressen

- 83 Autoren/Organisatoren