

Die TeXnische Komödie

dante

Deutschsprachige
Anwendervereinigung TeX e.V.

25. Jahrgang Heft 3/2013 August 2013

3/2013

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/DVD) 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-0.76.0-2013061708 (rev 4627), `format=lualatex 2013.7.8`, \TeX Live 2013 erstellt. Als Standard-Schriften kamen Linux Libertine, Linux Biolinum, Luxi Mono und XITS Math zum Einsatz.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 2500

Herausgeber: DANTE, Deutschsprachige Anwendervereinigung \TeX e.V.

Postfach 10 18 40

69008 Heidelberg

E-Mail: dante@dante.de (DANTE e.V.)

dtkred@dante.de (Redaktion)

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

Redaktion: Herbert Voß (verantwortlicher Redakteur)

Mitarbeit: Gert Ingold Eberhard Lisse Heiko Oberdiek

Christine Römer Volker RW Schaa Gert Seidl

Martin Sievers Dominik Waßenhoven

Redaktionsschluss für Heft 4/2013: 15. Oktober 2013

ISSN 1434-5897

Editorial

Liebe Leserinnen und liebe Leser,

diese Ausgabe widmet sich in dem Hauptbeitrag den Grundlagen von \TeX / \LaTeX ; Markus Kohm erklärt die Bedeutung des Zeichens »@« Dieses wird immer wieder in den diversen Mailinglisten, Foren und Tagungen hinterfragt. Roger Jud erklärt in seinem Beitrag, wie man in dem unter Windows häufig benutzen \LaTeX -Editor \TeX nicCenter die Autovervollständigung aktivieren kann. In einem weiteren Beitrag zeigt er die Anwendung von TikZ.

DANTE e.V. ist auf vielen Veranstaltungen präsent, um für die Sache \TeX und den Verein Werbung zu machen. Falk Hohlfeld ist dabei für DANTE e.V. regelmäßig unterwegs und teilt an zwei Beispielen seine Erfahrungen mit.

Mit dieser Ausgabe erhalten Sie auch die diesjährige DVD \TeX -Collection mit dem aktuellen \TeX Live 2013. Bevor die DVD in den Druck geht, gibt es \TeX Live sowohl zum Herunterladen als Teil der \TeX -Collection oder auch als Netzwerkinstallation. Dabei kann es zu Problemen kommen, wenn die Bekanntmachung dieser Tatsache schneller ist als die Aktualisierung der weltweit verteilten Spiegelserver. Diesmal war ich selbst betroffen. Das Installationsskript, welches man sich von <http://tug.org/texlive/> herunterladen kann, war bereits auf 2013 aktualisiert, aber das Archiv, aus welchem das Skript dann die \TeX Live Pakete holen wollte, eben noch nicht. »Living on the edge« nennen die Engländer solche Momente.

»Die \TeX nische Komödie« wird seit einiger Zeit ausnahmslos mit $\text{Lua}\LaTeX$ gesetzt, aber immer noch mit $\text{Bib}\TeX$, was der Tatsache geschuldet ist, dass wir per Skript alle Einzelbibliografien der Beiträge erstellen und dann die dabei erzeugte `bb1`-Datei über den Befehl `\input` einfach einlesen. Mit $\text{Bib}\LaTeX$ und der Anwendung von `Biber` ist das nun nicht mehr so trivial. Eine Lösung ist aber auf der nicht gerade kurzen Liste der Punkte, die in Zukunft erledigt werden sollen.

Ich wünsche Ihnen wie immer viel Spaß beim Lesen und verbleibe mit \TeX nischen Grüßen,

Ihr Herbert Voß

Hinter der Bühne

Vereinsinternes

Grußwort

Liebe Mitglieder,

Mitte des Jahres beginnt nicht nur die sommerliche Urlaubszeit, sondern es wird auch mit Hochdruck an der Fertigstellung der jeweils aktuellen $\text{T}_{\text{E}}\text{X}$ -Collection gearbeitet. Dank des großen Engagements eines eingespielten Teams halten Sie pünktlich mit dieser DTK die DVD in Händen. Diese wurde in diesem Jahr für DANTE e.V. und die anderen europäischen Benutzergruppen von uns produziert, nachdem die »Nebenkosten« für Versand, Steuern und Zoll in den vergangenen Jahren doch sehr hoch waren. Dadurch konnten die Herstellungskosten insgesamt deutlich gesenkt werden.

Ich danke allen Helfern, die zur Neuauflage 2013 beigetragen haben. Wir werden die Preisentwicklung anderer Medien wie USB-Sticks im Blick behalten und im nächsten Jahr neu entscheiden, ob es Anpassungsbedarf gibt.

Zusätzlich enthält diese Ausgabe der DTK eine Übersetzung des Begleitartikels zur $\text{T}_{\text{E}}\text{X}$ -Collection. Für alle Mac-Nutzer möchte ich auch auf <http://tug.org/mactex/> verweisen, wo Spezialitäten dieser $\text{T}_{\text{E}}\text{X}$ Live-Variante beschrieben sind.

Ebenfalls regelmäßig im Sommer laufen die Vorbereitungen für die Herbsttagung an. Sie finden in dieser Ausgabe der DTK die offizielle Einladung zur Mitgliederversammlung in Köln sowie den Aufruf, Vorschläge für das Vortragsprogramm einzureichen. Uwe Ziegenhagen als lokaler Organisator wird zudem ein spannendes Begleitprogramm zusammenstellen. Aufgrund der guten Erreichbarkeit und des zumindest in manchen Bundesländern verlängerten Wochenendes hoffe ich auf viele Teilnehmer und ein kurzweiliges Treffen.

Als DANTE e.V. gegründet wurde, gab es noch den »Eisernen Vorhang«, auch wenn dieser langsam durchlässig wurde. Heute leben wir in einem vereinigten Europa, worüber ich persönlich sehr froh bin. Gleichwohl gibt es natürlich auch für uns als Verein EU-bedingte Vereinheitlichungen und Regulierungen, die mehr Arbeit verursachen als sie (uns) Nutzen bringen. Dazu gehört wohl auch die Einführung des einheitlichen Euro-Zahlungsverkehrsraums (»Single Euro Payments Area«, kurz SEPA). Dieser schreibt unter anderem vor, dass wir ab dem 1. Februar 2014

unsere Mitgliedsbeiträge nicht mehr per Kontonummer/Bankleitzahl, sondern nur noch per IBAN/BIC einziehen dürfen. Zudem müssen »Einzugsermächtigungen«, die künftig »SEPA-Lastschriftmandat« heißen, ganz bestimmten Vorschriften genügen. Schließlich benötigt DANTE e.V. eine »Gläubiger-Identifikationsnummer« und eine möglichst komfortable und preiswerte Lösung für die künftigen Abbuchungsvorgänge. Änderungen an Formularen und bei der Vereinssoftware runden das Arbeitspaket ab.

Was bedeutet das nun konkret für Sie? Bestehende Einzugsermächtigungen können weiterhin verwendet werden, jedoch benötigen wir, wie oben beschrieben, künftig die europaweit gültigen IBAN/BIC-Angaben. Daher werden wir alle Mitglieder in den nächsten Monaten anschreiben und um Überprüfung der hinterlegten und von uns konvertierten Daten bitten. Für die Mitgliedsbeiträge 2014, die spätestens im Januar abgebucht werden, können wir noch auf das erprobte alte Verfahren zurückgreifen, so dass genügend Vorlauf bis zum ersten SEPA-Einsatz bleiben sollte. Zudem ist nicht auszuschließen, dass es zu weiteren Anpassungen des Verfahrens kommt, nicht zuletzt, weil laut Presseberichten insbesondere viele Vereine noch gar nicht mit der Umstellung begonnen haben.

Ich bedanke mich bei allen Beteiligten, insbesondere bei Karin Dornacher für die bereits geleisteten Schritte und Überlegungen. Es ist immer wieder schön zu sehen, dass uns auch solche Herausforderungen nicht aus der bürokratischen Bahn werfen. Zum Schluss möchte ich noch auf eine weitere Beilage zu dieser DTK hinweisen. Der Verlag Lehmanns Media, mit dem wir als Verein seit vielen Jahren zusammenarbeiten, stellt darin das »Thema ePub« vor. Dies ist sicherlich für einige nichts Neues, aber für andere ein spannendes Feld, das auch auf DANTE-Tagungen bereits Beachtung gefunden hat.

Nach diesen eher organisatorischen Informationen bleibt mir nur noch, Ihnen eine spannende und sommerliche Lektüre mit dieser Ausgabe der DTK zu wünschen, egal, ob im Urlaub oder auch zu Hause.

Herzlichst Ihr/Euer
Martin Sievers



(v. l.: Uwe Ziegenhagen, Manfred Lotz, Martin Sievers)



(v. l.: Volker RW Schaa, Uwe Ziegenhagen, Manfred Lotz)



(v. l.: Uwe Ziegenhagen, Manfred Lotz, Martin Sievers, Klaus Höppner)



(v. l.: Herbert Voß, Volker RW Schaa, Uwe Ziegenhagen)

Vorstandssitzung im Büro von DANTE e.V. (Fotos: Herbert Voß)

Einladung zur Herbsttagung 2013 und 49. Mitgliederversammlung von DANTE e.V.

Martin Sievers, Uwe Ziegenhagen

Liebe Mitglieder von DANTE e.V.,

hiermit laden wir Sie ganz herzlich zur Herbsttagung 2013 und 49. Mitgliederversammlung am 2. November 2013 in Köln ein. Diese Premierenveranstaltung richtet DANTE zusammen mit dem Regionalen Rechenzentrum (RRZK) und dem Department für Geowissenschaften der Universität zu Köln aus.

Die genaue Anschrift lautet:

Universität Köln
Ersatzbau Geowissenschaften
Greinstraße 6
50939 Köln

Für Tagung und Mitgliederversammlung ist folgender Zeitplan vorgesehen:

Freitag, 1. November, ab 19:00 Uhr: Vorabendtreff
Samstag, 2. November, 9:00 Uhr: Mitgliederversammlung
anschließend bis ca. 17:30 Uhr: Tagungsprogramm mit Vorträgen
ab 19:00 Uhr: Abendtreff
Sonntag, 3. November, ab 10:00 Uhr: Touristikprogramm

Die Tagesordnung der Mitgliederversammlung lautet:

1. Begrüßung und Tagesordnung
2. Bericht des Vorstands
3. Projektförderung/Erhöhung der Projektmittel
4. Verschiedenes

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

Detaillierte Informationen zur Tagung finden Sie unter <http://www.dante.de/events/Herbst2013.html>. Diese Seite wird im Laufe der nächsten Wochen fortlaufend aktualisiert. Die Teilnahme an der Tagung ist kostenfrei möglich, es wird

jedoch um vorherige Anmeldung bis zum 15. Oktober 2013 gebeten. Bitte nutzen Sie dazu das Formular auf der Tagungsseite.

Falls Sie sich aktiv am Vortragsprogramm beteiligen wollen, reichen Sie Ihren Vorschlag bitte schnellstmöglich per E-Mail an mv49@dante.de ein. Fügen Sie hierzu bitte eine Kurzzusammenfassung (Abstract) als Text- oder T\TeX -Datei in einem der üblichen Formate (L\AT\TeX , X\ET\TeX , Lua\T\TeX oder Con\T\TeX) bei. Die Vorträge sollten auf 30 Minuten (plus 10 Minuten Diskussion) ausgerichtet sein. Ausnahmen, insbesondere für Tutorien, sind möglich.

Für Fragen, Wünsche und Anregungen nutzen Sie bitte obige E-Mail-Adresse oder wenden Sie sich an

DANTE e.V.

Stichwort: Mitgliederversammlung von DANTE e.V.

Postfach 10 18 40

69008 Heidelberg

E-Mail (bevorzugt): mv49@dante.de

Wir freuen uns auf viele neue und alte Gesichter in Köln, das auch mit einem reizvollen touristischen Beiprogramm lockt und immer eine Reise wert ist.

Mit freundlichen Grüßen

Martin Sievers (Vorsitzender DANTE e.V.)

Uwe Ziegenhagen (Köln)



Foto: Rolf Heinrich, Köln

T_EX-Theatertage

Messegeflüster

Falk Hohlfeld

Die sich dem mittleren Alter nähernde Dame »DANTE« hat sich auf dem Berliner Linuxtag und der FuXCon in Frankfurt wacker geschlagen. Beide Veranstaltungen konnten nicht widersprüchlicher sein. Die Linuextage sind eine allgemein anerkannte Größe und sind obwohl oder gerade weil viel Ehrenamt dahinter steckt, eine sich sehr professionell darstellende Einrichtung. Hier ist in der Regel alles vertreten, was Rang und Namen in der weiten OpenSource- und Linuxwelt hat.

Die FuXCon fand zum ersten Mal statt. Ein zartes Pflänzchen, das sich mit viel Geduld bestimmt weiter entwickeln wird. Was unbedingt hervorzuheben ist, ist die Tatsache, dass sich die Organisation in Frankfurt auf keinen Fall hinter Berlin verstecken muss. Die Gruppen, die sich vorgestellt haben, sind Basis aller OpenSource- und Linuxaktivitäten im Rhein-Main-Gebiet. Natürlich brauchte man auch ein paar Sponsoren. Hier waren unter anderen S&P Media, O'Reilly, Namics und msg anwesend.

DANTE e.V. war in Berlin mit Herbert Voß, Martin Schmidt, Martin Wilhelm »Moss« Leidig, Harald König, Uwe Ziegenhagen und Sven Neumann vertreten. In Frankfurt war die Crew nicht ganz so groß, hier waren Kalle Ohnemus, Manfred Lotz und Jürgen Rosenow für DANTE e.V. unterwegs.

Interessanterweise waren die Altersstruktur und die im Gespräch behandelten Themenbereiche bei beiden Veranstaltungen sehr ähnlich. Die größte Gruppe stellen die unter Dreißigjährigen mit ca. 40 %. Die zwei Gruppen der unter 40- und unter 50-jährigen sind gemeinsam mit 40 % ebenso stark wie die ganz Jungen. Aber auch die über 50- und über 60-jährigen sind mit jeweils 10 % der Gesprächssuchenden präsent.

In allen Gesprächen wurde versucht, neben den T_EX-Fragen auch unseren Verein vorzustellen, so er nicht schon bekannt war. Hier bot sich natürlich das Angebot der Schnuppermitgliedschaft als Fortsetzung des Gespräches an. In Berlin konnten insgesamt fünf neue »Freunde« gewonnen werden.

Der Bekanntheitsgrad von \TeX war sehr hoch. Dass jemand noch gar nichts von \TeX oder seinen Derivaten gehört hatte, war ganz selten. Trotz des hohen Bekanntheitsgrades von \TeX wurden doch die allgemeinen Fragen, »wie funktioniert \TeX allgemein«, »Einsatzgebiete«, »Geschichte der Entwicklung«, »Lernkurve« und »weiterführende Literatur« am häufigsten gestellt. An zweiter Stelle wurde viel über den Verein DANTE e.V. gesprochen; Aufgaben, Mitgliedschaft, Publikationen und Tätigkeiten bildeten hier die Schwerpunkte der Standbetreuung. Es wurde wirklich nicht langweilig. Auf den Plätzen drei und vier der Häufigkeit von angesprochenen Themen waren die Bereiche Schriften, Pakete, Formate und Einbindung, sowie eine Vielfalt von Fragen zur Gestaltung und Bearbeitung von Bibliografien.

Etwas fundamentalistisch wurde die Diskussion, wenn es um die verschiedensten Editoren ging, die natürlich auch in den bekannten Betriebssystemen unterschiedlich vorhanden sind. Hier hat selbstverständlich jeder seinen ganz persönlichen Favoriten. Bei der Vielzahl der Angebote konnte man nur eine grobe Empfehlung aussprechen. Die über Beamer vorgeführte Version von »Gummi« fand große Beachtung. Es ist ein relativ neuer und einfacher Editor für Linux, der sich durch seine Zweifenstertechnik (eine Seite Code, die andere Seite PDF) gut bedienen lässt. Die Kompilierung des Quelltextes erfolgt innerhalb eines einstellbaren Zeitraumes automatisch. Zum anderen wurde eine Variante über Emacs mit Einsatz von vi/vim und diversen Plugins und Arbeit in verschiedenen Modi diskutiert. Ein sehr interessantes, auch zukunftsweisendes Thema war die Frage, wie man von \TeX zu ePub und allen ähnlichen Formaten kommt. Reines PDF stellt ja kein Problem dar, aber die Umwandlung von \TeX in die verschiedenen Formate der E-Reader ist noch nicht zufriedenstellend gelöst. Weitere Themen, jetzt ohne eine Reihenfolge, waren Fragen zu Fußnoten, automatisches Anpassen von Bildgrößen in Präsentationen, Mathematiksatz, Makroeinbindung in \TeX und Notensatz in \TeX -Dokumenten.

In Frankfurt waren nun wesentlich weniger Standbesucher, aber die Themenvielfalt war ähnlich wie in Berlin, vom Einzelnen zum Besonderen.

Ein paar abschließende Bemerkungen. Unter allen Besuchern in Berlin und Frankfurt waren Frauen sehr stark unterrepräsentiert. Es waren weniger als 5 %. In Frankfurt war außer den Veranstaltern keine Frau, die sich für \TeX interessierte. Bei den Informatikstudiengängen ist in den letzten Jahren dafür eine starke Zunahme, bis zu 50 % eines Jahrganges, von Studentinnen zu vermerken. Ein offensichtlicher Widerspruch, der so noch nicht erklärbar ist.

Wenn man ein paar Veranstaltungen wie die jeweiligen Linuextage, Messen u. a., besucht hat, kennt man sie schon, die »Dauerbesucher« und diejenigen, die teilweise immer dieselben Fragen stellen. Aber auch hier heißt es, immer freundlich sein, schließlich sind ja alle Beteiligten ehrenhafte \TeX -Löwinnen und -Löwen.

Nach den Linuxtagen ist vor den Linuxtagen. Die nächsten größeren Treffen sind die FrosCon in St. Augustin, der Bayerische Sommerstammtisch in Nürnberg und die OpenRheinRuhr in Oberhausen.



Linuxtag (Fotos: Sven Naumann)



FrosCon (Foto: Uwe Ziegenhagen)

Die DVD mit der T_EX-Collection 2013¹

Herausgeber der T_EX-Collection

T_EX-Collection ist die Bezeichnung für die Zusammenstellung der Software, die jedes Jahr an die T_EX-Benutzergruppen verteilt wird. Bitte würdigen Sie, dass sich die TUG (T_EX Users Group) oder die Gruppe ihrer Wahl (<http://tug.org/usergroups.html>) dafür engagiert. Sie können auch mit einer Spende (<https://www.tug.org/donate.html>) diese Bemühungen unterstützen.

Alle diese Projekte werden ausschließlich von Freiwilligen durchgeführt. Wenn Sie bei der Entwicklung, dem Testen, der Dokumentation, etc. helfen möchten, besuchen Sie bitte die Projektseiten, dort bekommen Sie weitere Informationen, wie Sie etwas dazu beitragen können.

Vielen Dank an alle Beteiligten.

proT_EXt (<http://tug.org/protext>)

proT_EXt umfasst ein T_EX-System für MS Windows, das auf MiK_TE_X basiert (<http://miktex.org>), sowie die entsprechenden Editoren T_EXStudio (früher bekannt als T_EXmakerX) (<http://texstudio.sf.net>).

In der aktuellen Fassung bringt proT_EXt jetzt ein Installationsprogramm (Setup.exe) mit. Daneben gibt es eine detaillierte Anleitung, die Sie durch die Installation führt (und die Sie wahrscheinlich aufgrund der einfachen Installation nie benötigen werden).

Derzeit hat proT_EXt Englisch, Deutsch und Französisch als mögliche Installations-sprachen. Freiwillige für das Anfertigen zusätzlicher Übersetzungen sind höchst willkommen.

MacT_EX (<http://tug.org/mactex>)

MacT_EX ist ein T_EX-System für MacOSX, es installiert T_EXLive und zusätzliche Mac-spezifische Tools. MacT_EX 2013 läuft sowohl mit Intel- als auch PowerPC-Maschinen und erfordert mindestens MacOSX 10.5 (Leopard). Es läuft mit MacOSX Leopard, Snow Leopard, Lion und Mountain Lion. Das Paket installiert das vollständige T_EXLive 2013, Ghostscript 9.07, die convert-Utility von ImageMagick 6.8.3-3 und die aktuellen Versionen von BibDesk, L^AT_EXiT, T_EXLive Utility, TeXShop und T_EXworks,

¹ Übersetzung durch Christine Römer

sowie die » \TeX Dist Preference Pane«, die den Benutzern erlaubt, einfach zwischen verschiedenen \TeX -Distributionen zu wechseln. Nach der Installation ist der Pfad für die Shells gesetzt, und alle Applikationen und das System sind betriebsbereit.

Die Technik zur Installation dieser Software wurde 2013 auf eine einfachere zwei- oder dreiteilige Methode geändert, die mehr Rückmeldungen bereitstellt. Zuerst starten Sie als Benutzer ein Terminal (aus `/Applications/Utilities`) und kopieren zwei Textzeilen aus diesem Programm von einem kurzen *Mac \TeX Install Part 1* Dokument. Dieser Schritt installiert ein komplettes \TeX Live von der DVD, ohne Fragen zu stellen. Zweitens führt der Benutzer ein Programm aus, das alles Andere installiert und das System konfiguriert. Der dritte Teil ist nur für Benutzer des älteren Leopard oder Snow Leopard nötig; er fügt ein GUI-Frontend ein, das auf diesen Systemen läuft.

Die DVD beinhaltet auch Mac \TeX tras (<http://tug.org/mactex/mactextras.html>) und enthält viele weitere Elemente, die separat installiert werden können. Die Software, die ausschließlich auf Tiger (MacOSX 10.4) läuft, wurde aus der 2013-DVD zusammen mit der Software, die vom Mac \TeX -Installer installiert wurde, entfernt. Die Hauptkategorien sind: Bibliografieprogramme, alternative Editoren, Satzsetzer und Previewer; Gleichungsedatoren; DVI- und PDF-Previewer und Rechtschreibprüfer.

\TeX Live (<http://tug.org/texlive>)

\TeX Live ist ein umfassendes, plattformübergreifendes \TeX -System. Es bietet Support für die meisten Unix-ähnlichen Systeme, einschließlich GNU/Linux und MacOSX, sowie für MS Windows. Wichtige für die Benutzer sichtbare Veränderungen in 2013 sind:

- Der `texmf`-Baum wurde in `texmf-dist` eingefügt; Sprachen-Kollektionen wurden zusammengeführt.
- Zu \XeTeX : Siehe den Artikel von Khaled Hosny »What is new in XETEX 0.9999?« in TUGboat, Volume 34 (2013), No. 2.
- Zu \LuaTeX : Wurde aktualisiert auf Lua 5.2; eine neue Bibliothek zum Verarbeiten von externen PDF-Seiteninhalt.
- Zu MetaPost: Nativer Support für eine PNG-Ausgabe und Fließkommaarithmetik (IEEE double) wurde aufgenommen.
- Zu `xdvi`: Nutzt nun FreeType anstelle von `t1lib` für das Rendering.
- Zu `tlmgr`: Neue `pinning`-Aktionen zur einfachen Konfiguration von multiplen Repositories.
- Zu den Plattformen: `armhf-linux`, `mips-irix`, `amd64-netbsd`, `i386-netbsd` wurden hinzugefügt oder wiederbelebt; entfernt wurde `powerpc-aix`. Um Platz

zu sparen, wurden außerdem einige Plattformen auf der DVD weggelassen, sie können in der Regel aber über das Internet installiert werden.

Weitere Details gibt es im \TeX Live-Handbuch und auf den Webseiten.

CTAN (<http://www.ctan.org>)

CTAN ist das »Comprehensive \TeX Archive Network«, eine Reihe von Servern macht die \TeX -Software weltweit öffentlich verfügbar.

Wie üblich wurde der Abzug von CTAN über den deutschen Knoten (<http://dante.ctan.org>) erstellt und die anderen Komponenten der Kollektion ausgelassen. Der CTAN-Abzug ist für die \TeX Users Group Mitglieder auf der » \TeX Users Group members area«, <https://www.tug.org/members>, verfügbar.



Bretter, die die Welt bedeuten

Was ist eigentlich: die Besonderheit des @-Zeichens in Befehlsnamen?

Markus Kohm

Heutzutage finden viele Anwender ihre Informationen zu \TeX und \LaTeX nicht mehr nur in Büchern, sondern irgendwo im Internet. Dabei werden oftmals von den Helfern Begriffe und Dinge verwendet, bei denen das Wissen um deren Bedeutung einfach vorausgesetzt wird. »*Was ist eigentlich*« soll in lockerer Folge ein paar Erklärungen liefern. Dieses Mal geht es um das Zeichen »@« innerhalb von Befehlsnamen.

Gerade bei Fragen, deren Lösung nicht in einer einfachen Verwendung der von einer Klasse oder einem Paket bereitgestellten Möglichkeiten besteht, wird der Leser von Internetforen häufig mit Befehlen konfrontiert, die ein »@« im Namen haben. Dabei drängt sich der Verdacht auf, dass diese nicht wie andere Befehle zu verwenden sind. Dieser Artikel wird den Verdacht nur teilweise bestätigen.

Grundlagen

Vorab ist zu sagen, dass der Titel des Artikels zwar griffig aber etwas ungenau ist. Befehle im Sinne einer deklarativen Programmiersprache kennt \TeX nicht. Etwas vereinfacht kennt \TeX Primitive, Makros, Token und Register. Register sind beispielsweise Längen (engl. *dimensions*), Abstände (engl. *skips*), Zähler (engl. *counters*), Kästen (engl. *boxes*) und einige mehr. Token sind die kleinsten internen Verarbeitungseinheiten. Zeichen werden beim Einlesen des Quellcodes zu Token. Makros expandieren zu Token. Makros sind Verarbeitungseinheiten, die aus Token zusammengesetzt sind. Wir kennen diese auf \LaTeX -Ebene meist als Anweisungen. Sie können jedoch auch vergleichbar zu Variablen verstanden werden. Wichtig ist zu wissen, dass die Token, aus denen ein Makro besteht, bereits bei der Definition des Makros gebildet werden. Lediglich die Argumente, die ein Makro haben kann, werden erst beim Lesen der Argumente in Token umgewandelt und bei der Verwendung des Makros eingesetzt. Primitive sind sozusagen die Makros, die bereits als Teil der Sprache \TeX selbst bereitgestellt werden. Im Unterschied zu tatsächlichen

Makros expandieren sie nicht zu anderen Token, sondern zu einem Token, nämlich sich selbst.

Dazu ein Beispiel:

```

1 \documentclass{article}
2 \newcommand*{\test}[1]{Ein Test.}
3 \begin{document}
4 \test
5 \end{document}

```

Hier wird zunächst per `\newcommand` das Makro `\test` definiert. Beim Einlesen der Definition werden aus »Ein Test.« die folgenden Token gebaut, wobei im Index jeweils die Art des Tokens steht und das `_` für ein Leerzeichen steht:

$E_{\text{Buchstabe}}$ $i_{\text{Buchstabe}}$ $n_{\text{Buchstabe}}$ $_ \text{Leerzeichen}$ $T_{\text{Buchstabe}}$ $e_{\text{Buchstabe}}$ $s_{\text{Buchstabe}}$
 $t_{\text{Buchstabe}}$ $_ \text{Anderes}$

Bei der Verwendung von `\test` im Dokument werden genau diese Token eingesetzt. Dieses Einsetzen wird als Expansion bezeichnet.

Wie aus dem Beispiel hervor geht, kennt \TeX unterschiedliche Arten von Token. Das beginnt aber nicht erst bei den Token, sondern bereits beim Einlesen der Zeichen. Die Art eines Zeichens bestimmt, welches Token daraus gebildet wird. So wie es Buchstaben-Token und Anderes-Token gibt, gibt es Buchstaben-Zeichen und Anderes-Zeichen. Zeichen werden jedoch nicht 1:1 auf Token abgebildet. So werden bei der Makrodefinition

```

1 \newcommand*{\test}{\mbox{Test}}

```

keine 11 Token gespeichert, sondern ein Befehl-Token mit dem Inhalt »mbox« und ein Ausgeglichen-Text-Token, das wiederum aus den vier Buchstaben-Token »T«, »e«, »s« und »t« besteht. Wichtig ist dabei für uns das Befehl-Token. Dabei spielt es übrigens keine Rolle, ob das Befehl-Token nun für ein Makro, ein Register oder eine Primitive steht.

Bei \TeX gehören normalerweise die Buchstaben A bis Z und a bis z zur Zeichenart Buchstabe. Das Zeichen `\<<` gehört zur Zeichenart Escape. `\{<`, `\}<`, `\$<`, `\&<`, `\#\<`, `\^\<`, `\~<`, `_<`, `\%<` und verschiedene Steuerzeichen haben eigene Zeichenarten, die an dieser Stelle von geringer Bedeutung sind. Das Leerzeichen hat ebenfalls eine eigene Zeichenart Leerzeichen. Alle übrigen Zeichen gehören zu der Zeichenart Anderes. Dazu gehört auch das `\@<`-Zeichen, wenn sie nicht durch ein anderes Paket beispielsweise der Zeichenart Aktiv zugeordnet wurden. Ein Beispiel dafür wäre das Anführungszeichen bei Verwendung von `babel` mit Option `ngerman`.

Bildung von Befehl-Token

Befehl-Token werden beim Einlesen aus Zeichen wie in Abbildung 1 dargestellt erzeugt. Der »nein«-Ausgang bei »Zeichenart Aktiv?« wurde bewusst offen gehalten. Was an der Stelle geschieht, ist im Rahmen dieses Artikels unerheblich.

Wie der Abbildung zu entnehmen ist, wird ein Befehl-Token also entweder aus einem Zeichen der Zeichenart Aktiv oder aus einem Zeichen der Zeichenart Escape und einer Reihe von Zeichen der Zeichenart Buchstabe oder aus einem Zeichen der Zeichenart Escape und genau einem Zeichen einer anderen Zeichenart gebildet.

Im vorherigen Abschnitt wurde erklärt, dass das »@«-Zeichen normalerweise zur Zeichenart Anderes gehört. Demnach kann damit also normalerweise nur ein Befehl-Token gebildet werden, wenn es unmittelbar auf ein Zeichen der Zeichenart Escape folgt. Das ist normalerweise nur das »\«-Zeichen. Der Befehl \@ ist bei \LaTeX normalerweise als `\spacefactor \@m` vordefiniert. Jedes Zeichen, das auf \@ folgt, wird gemäß Abbildung 1 nicht mehr als Teil des Befehlsnamens erkannt. Das erklärt nun auch, warum bei versehentlicher Verwendung von »@« am vermeintlichen Anfang von Befehlsnamen häufig die Fehlermeldung

! You can't use '\spacefactor' in vertical mode.

auftaucht.

Ebenso sind die Fehlermeldungen wegen undefinierter Anweisungen erklärlich, wenn versehentlich »@« vermeintlich innerhalb eines Befehlsnamens verwendet wird. Nach dem Diagramm aus Abbildung 1 kann »@« gar nicht innerhalb eines Befehlsnamens stehen. Einige Leser werden nun verwundert ausrufen, dass es aber doch solche Befehle gebe. Darauf kommen wir gleich.

Befehle mit »@« im Namen

Bereits Donald Knuth entschied bei der Erstellung seiner grundlegenden Befehle des plain \TeX -Formats, dass Befehlsnamen nur aus Buchstaben zu bestehen haben. Als Leslie Lamport in den 80er Jahren des letzten Jahrhunderts die Urform von \LaTeX entwickelt hat, vermisste er allerdings eine klare Trennung des Namensraums für die Benutzerebene und für die darunter vor dem Benutzer verborgen gehaltene Entwicklerebene. Daher entschied er, dass innerhalb des \LaTeX -Kerns und innerhalb der Stil-Dateien »@« die Zeichenklasse Buchstabe haben soll und die Befehle der Entwicklerebene allesamt mindestens ein »@« im Namen haben sollen.²

Damit begründete er eine reine Konvention:

²Wir erinnern uns, dass die Beschränkung, dass »@« nicht in Befehlsnamen vorkommen kann, keine Eigenschaft des Zeichens selbst, sondern der Zeichenart Anderes ist, der das Zeichen normalerweise angehört.

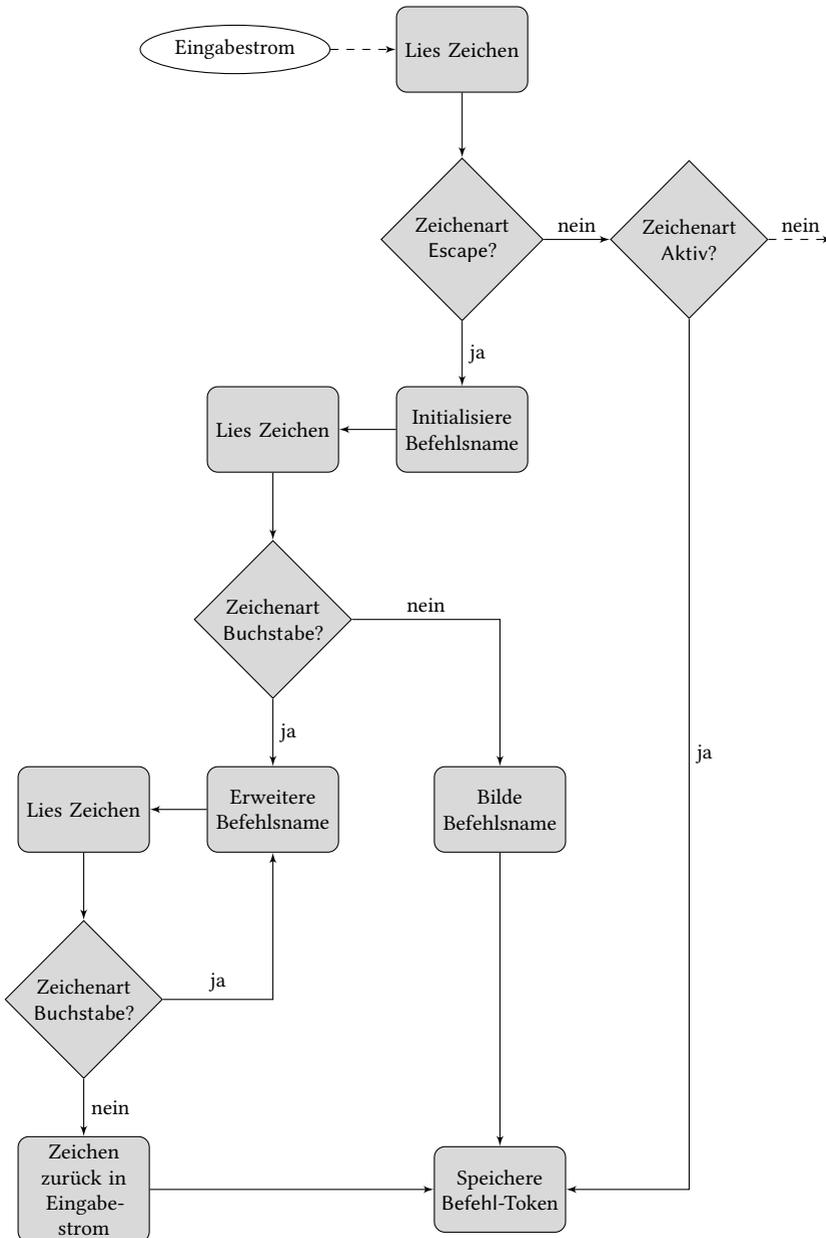


Abbildung 1: Erzeugung eines Befehlstokens beim Einlesen des Quellcodes

- Auf *Benutzerebene* bestehen Befehlsnamen nur aus Buchstaben.
- Auf *Entwicklerebene* enthalten Befehlsnamen immer ein »@«.

Es handelt sich hier also keinesfalls um eine technische Notwendigkeit, sondern um eine Übereinkunft.

Damit Benutzer nicht versehentlich Befehle der Entwicklerebene umdefinieren und damit sie nach Möglichkeit die ihnen zugeordnete Benutzerebene auch einhalten, wurden zur Durchsetzung der ersten Konvention zwei technische Maßnahmen ergriffen:

- Innerhalb von Dokumenten gehört »@« weiterhin zur Zeichenart Anderes.
- Beim Lesen von Stil-Dateien und einigen Hilfsdateien gehört »@« zur Zeichenart Buchstabe.

Bei Einführung von $\LaTeX 2_\epsilon$ wurden sowohl die Konventionen als auch die Maßnahmen beibehalten, wobei die zweite Maßnahme nun lautet:

- Beim Lesen von Klassen-, Paket- und einigen Hilfsdateien gehört »@« zur Zeichenart Buchstabe.

Es wurden keine technischen Maßnahmen ergriffen, dass Klassen- und Paketautoren auch wirklich immer ein »@« in die Befehle der Entwicklerebene einbauen. Stattdessen gehört diese Konvention zur Eigenverantwortung der Entwickler.

Aber bereits bei der Entwicklung des \LaTeX -Kerns zeigte sich ein weiteres Problem. Einige Anweisungen der Entwicklerebene sollten auch anderen Entwicklern zugänglich sein. So stellt der \LaTeX -Kern beispielsweise mit `\if@twoside` einen Schalter bereit, den Klassenautoren unbedingt verwenden und gegebenenfalls sogar ändern sollten. Auch die Befehle zur Definition von Gliederungsbefehlen, wie `\@startsection` gehören dieser Schicht an. Man könnte diese Schicht die *Entwicklerschnittstelle* nennen.

Unterhalb der Entwicklerschnittstelle gibt es aber auch Anweisungen, die nur für den internen Ablauf benötigt werden. Würde man all diese der Entwicklerschnittstelle zuordnen, wäre die Funktion einer Klasse oder eines Pakets weitgehend unveränderlich festgeschrieben. Nicht einmal Fehlerkorrekturen wären dann noch in jedem Fall möglich. Diese Anweisungen zählen wir daher zu den *internen Anweisungen*. Leider gibt es bei \LaTeX keine zuverlässige Trennung von internen Anweisungen und Anweisungen der Entwicklerschnittstelle, ähnlich der technisch durchgesetzten Konvention zur Trennung von Entwicklerebene und Anwenderebene.³

³ Bei $\LaTeX 3$ hat man aus diesem Fehler übrigens gelernt. Dort ist das ein wenig anders.

Was nun aber, wenn ein Entwickler für seine Klasse oder sein Paket auf Anweisungen des \LaTeX -Kerns, oder aus Klassen oder Paketen zurückgreifen muss, die ein »@« im Namen tragen, also der Entwicklerebene zugeordnet sind. Woher soll er wissen, ob sie Teil der Entwicklerschnittstelle sind oder zum internen Teil gehören, von dem er besser die Finger lassen sollte?

Beim \LaTeX -Kern ist die Frage pragmatisch zu beantworten. Da die Funktion von $\LaTeX 2_\epsilon$ weitgehend festgeschrieben ist, können alle internen Anweisungen ohne große Sorge verwendet werden. Allerdings sollte man die internen Anweisungen möglichst so verwenden, dass insbesondere andere Pakete nicht gestört werden und durch andere Pakete keine Störung zu erwarten ist. Eine Unsitte ist beispielsweise, interne Anweisungen komplett neu zu definieren, wenn tatsächlich nur ein Patch (siehe Paket `etoolbox`) oder eine erweiternde Definition notwendig ist. Es ist also besser

```

1 \newcommand*{\foo@saved@secntformat}{}
2 \let\foo@saved@secntformat\@secntformat
3 \renewcommand*{\@secntformat}[1]{%
4   \typeout{#1: \csname the#1\endcsname}%
5   \foo@saved@secntformat{#1}%
6 }
```

als mit dem Wissen, dass `\@secntformat` im \LaTeX -Kern als

```

1 \def\@secntformat#1{\csname the#1\endcsname\quad}
```

definiert ist,

```

1 \renewcommand*{\@secntformat}[1]{%
2   \typeout{#1: \csname the#1\endcsname}%
3   \csname the#1\endcsname\quad
4 }
```

zu schreiben.

Für Pakete und Klassen, die durchaus noch Veränderungen unterworfen sein können, ist die Frage weit schwerer zu beantworten. Als erstes ist zu prüfen, ob es nicht bereits eine Lösung auf Anwenderebene gibt.

Wenn man dann doch eine Befehlssequenz mit »@« im Namen verwendet, sollte man in der Dokumentation zu dieser unbedingt nachlesen, ob sie für Paket- und Klassenautoren ausdrücklich freigegeben ist. Falls dies nicht der Fall ist, muss man davon ausgehen, dass es sich um interne Anweisungen oder Register handelt. In diesem Fall sollte man sich zweimal überlegen, ob man darauf zurückgreift. Falls man es tut, ist unbedingt zu dokumentieren, zu welchem Zweck man sie verwendet und aus welcher Quelle sie stammen. Ein Kommentar mit einer Warnung, dass dies bei zukünftigen Versionen der entsprechenden Klasse oder des entsprechenden Pakets zu Problemen führen kann, ist dann auf jeden Fall angebracht. Auch eine

Absicherung mit `\CheckCommand` (siehe [1]) kann in solchen Fällen nicht schaden. Noch besser ist es, wenn man sich mit dem Autor des Pakets oder der Klasse, aus der die interne Anweisung stammt, in Verbindung setzt. Möglicherweise lässt sich gemeinsam eine definierte Schnittstelle finden.

In dem Beispiel oben wird auch noch ein weiteres Problem von \TeX deutlich. Es gibt keine Abtrennung der Namensräume für Entwickler. Daher ist es auch nicht sinnvoll, wenn Entwickler für interne Befehlssequenzen Namen verwenden, die einfach nur einen allgemein naheliegenden Präfix wie »`\my@`« oder »`\orig@`« haben. Stattdessen sollte man für interne Anweisungen immer einen Präfix verwenden, der möglichst eindeutig dem eigenen Paket zugeordnet werden kann und bei dem die Wahrscheinlichkeit, dass ein anderer denselben Präfix verwendet, möglichst gering ist. Der Präfix »`foo@saved@`« taugt im Übrigen auch nur für Beispiele und nicht für eine konkrete Verwendung!

Verwendung von Entwickleranweisungen mit »@« durch den Anwender

Anwender sollten selbstverständlich zunächst die Anleitung studieren, um möglicherweise eine Lösung auf dieser Ebene zu finden. Auch die Einbeziehung anderer Pakete, die eine solche Lösung anbieten, sollte der Verwendung von Entwickleranweisungen vorgezogen werden. Sollte all dies ausscheiden, so sollte er wie am Ende des vorherigen Abschnittes für Entwickler erklärt vorgehen.

Damit der Anwender die Anweisungen der Entwicklerebene überhaupt verwenden kann, benötigt er aber noch einen Kunstgriff. Er muss nämlich dafür sorgen, dass »@« als Teil des Befehlsnamens erkannt wird. Dafür bietet der \LaTeX -Kern in weiser Voraussicht die Anweisungen `\makeatletter` und `\makeatother`. Mit der ersten ordnet man das At-Zeichen, also »@«, der Zeichenart Buchstabe (engl. *letter*) zu. Danach darf man also »@« in Befehlsnamen verwenden. Am Ende sollte man nicht vergessen mit `\makeatother` das At-Zeichen, also »@«, wieder der Zeichenart Anderes (engl. *other*) zuzuordnen.

Ein häufiger Fehler von Anwendern ist übrigens, dass sie den Hinweis aus dem ersten Abschnitt nicht beachten, der da lautete: *»Wichtig ist zu wissen, dass die Token, aus denen ein Makro besteht, bereits bei der Definition des Makros gebildet werden. Lediglich die Argumente, die ein Makro haben kann, werden erst beim Lesen der Argumente in Token umgewandelt und bei der Verwendung des Makros eingesetzt.«* Damit wäre also

```

1 \newcommand*{\Test}[1]{%
2   \typeout{#1: \csname the#1\endcsname}%
3   \makeatletter
4   \@seccntformat{#1}%

```

```

5 \makeatother
6 }

```

fälsch. Zwar wird vor `\@secntformat` mit der Anweisung `\makeatletter` das »@« der Zeichenart Buchstabe zugeordnet. Allerdings wird `\@secntformat` bereits beim Lesen des Definitionstextes, der ja ein Argument von `\newcommand` ist,⁴ in Token umgewandelt. Zu dem Zeitpunkt gehört »@« aber noch zur Zeichenart Anderes und damit funktioniert die Definition nicht mehr. Korrekt ist daher:

```

1 \makeatletter
2 \newcommand*\Test}[1]{%
3   \typeout{#1: \csname the#1\endcsname}%
4   \@secntformat{#1}%
5 }
6 \makeatother

```

Es gibt auch noch eine zweite Möglichkeit, Befehlsnamen mit anderen Zeichen als Buchstaben zu verwenden. Das ist die \TeX -Primitive:

```
\csname <Befehlsnamencode>\endcsname
```

Dabei wird `<Befehlsnamencode>` zunächst expandiert und die Expansion dann als Befehlsname aufgefasst. Wir sehen das in obigen Beispielen häufiger beispielsweise für die Bildung eines Makros `\the<foo>` aus `the` und dem Argument der Anweisung `\Test`. Damit könnte man also das vorherige Beispiel auch als

```

1 \newcommand*\Test}[1]{%
2   \typeout{#1: \csname the#1\endcsname}%
3   \csname @secntformat\endcsname{#1}%
4 }

```

ganz ohne `makeatletter` und `makeatother` schreiben.

Schlussbemerkung

Auch wenn in diesem Artikel erklärt wurde, wie man Befehle und Register mit »@« im Namen verwenden kann, soll zum Schluss noch einmal darauf hingewiesen werden, dass man dies nur mit Bedacht tun sollte. In der Regel hat es einen guten Grund, dass die Verwendung dieser Befehle mit Hindernissen belegt ist. Wenn es also eine andere Möglichkeit gibt, dann solle man unbedingt diese andere Möglichkeit nutzen. Das schließt auch die Verwendung zusätzlicher Pakete ein.

Das Ablaufdiagramm in Abbildung 1 weist übrigens eine Ungenauigkeit auf. Das Zurückstellen von Zeichen in den Eingabestrom gilt nicht für Zeichen der Art Leerzeichen. Aber das ist Thema für einen weiteren Artikel der Reihe »*Was ist eigentlich*«.

⁴ Eigentlich ist er ein Argument von `\def`.

Literatur

- [1] The \LaTeX 3 Project: $\LaTeX 2_{\epsilon}$ for class and package writers; Feb, 2006; CTAN: macros/latex/doc/cslsguide.pdf; zuletzt besucht: 2013-06-06; Bestandteil jeder \LaTeX -Distribution.

Autovervollständigung mit \TeX nicCenter

Roger Jud

In diesem Beitrag soll gezeigt werden, wie man die Autovervollständigung von \TeX nicCenter benutzt und anpasst.

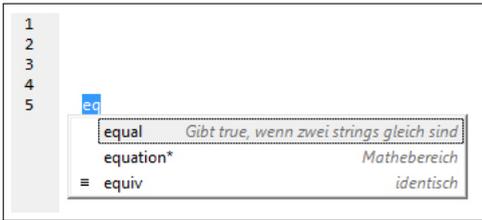
Problemstellung

Vom Standpunkt des Tipp-Aufwands her betrachtet ist die Erstellung und Bearbeitung von \LaTeX -Dokumenten selbst für erfahrene Benutzer mühsam, da man die verwendeten \LaTeX -Befehle einerseits auswendig wissen und diese andererseits eintippen muss. Insbesondere der Backslash »\«, welcher im Allgemeinen \LaTeX -Befehlen vorangeht, sowie geschweifte und eckige Klammern, welche obligatorische bzw. optionale Argumente umschließen, sind auf der Tastatur nur schwer erreichbar. Außerdem muss zu einem `\begin{umgebung}` auch immer ein entsprechendes `\end{umgebung}` eingegeben werden. Eine Autovervollständigung (engl. autocompletion) unterstützt den Autor bei sämtlichen obigen Problemen.

Autovervollständigung in \TeX nicCenter

\TeX nicCenter (<http://www.texniccenter.org>), ein \LaTeX -Editor für das Betriebssystem Windows, welcher aktuell in der Version 2 Beta 1 vorliegt, verfügt über eine entsprechende Autovervollständigung. Tippt man in \TeX nicCenter die ersten zwei oder mehr Buchstaben eines Befehls ein und drückt anschließend die Ctrl- (bzw. Strg-) und die Leertaste gleichzeitig, so erscheint ein entsprechendes Autovervollständigungsfenster (siehe Screenshot unten) oder der Befehl wird automatisch vervollständigt. Letzterer Fall tritt ein, wenn nur ein Befehl mit den entsprechenden Anfangsbuchstaben existiert.

Mit den Pfeiltasten lässt sich nun der gewünschte Befehl auswählen und mit der Eingabe-Taste einfügen. Alternativ kann dazu auch die Maus verwendet werden.



Anpassung und Ergänzung der Autovervollständigung

Die Befehlsnamen werden den Autovervollständigungsdateien entnommen, die sich im Ordner `\texniccenter\packages` befinden. \TeX nicCenter bringt von Haus aus drei dieser xml-Dateien mit: `base.xml`, `math.xml` und `TeX.xml`. Diese Autovervollständigungsdateien können vom Benutzer mit neuen Befehlen ergänzt werden, nicht benötigte Befehle können entfernt werden. Ebenso können im Ordner `\packages` weitere solcher Dateien erstellt werden. Am besten kopiert man dazu eine bestehende Datei und ändert diese ab. Wichtig ist, dass die Änderungen an den Autovervollständigungsdateien erst nach einem Neustart von \TeX nicCenter wirksam werden.

Aufbau einer Autovervollständigungsdatei

Der folgende Code zeigt beispielhaft eine Autovervollständigungsdatei, in der die Umgebung `equation*` sowie die Befehle `\frac` und `\alpha` definiert sind.

```

1 <?xml version="1.0" encoding="ISO-8859-1" standalone="no"?>
2 <!collection version="1" xmlns:txclxp="http://schemas.ToolsCenter.org/
  ↪ TeXnicCenter/LaTeXCommands.xsd">
3   <!package name="mathematik">
4     <!environment name="equation*"
5       parameters="0"
6       desc="Mathebereich"
7       expafter="}&#xA;&#x9;&#xA;\end{equation*}"
8       expbefore="\begin{"/>
9
10    <!environment name="frac{}{}"
11      parameters="0"
12      desc="Bruch"
13      expafter=""
14      expbefore="\"/>
15  </!package>
16
17  <!package name="griechische Buchstaben">
18    <!environment name="alpha"
19      parameters="0"
20      icon="symbols.bmp"
21      index="0"

```

```

22 desc="alpha"
23 expafter=""
24 expbefore="\ " />
25 </lxPackage>
26 </lxCollection>

```

Die Zeilen 1 und 2 sowie 26 müssen am Anfang bzw. am Ende einer jeden Autovervollständigungsdatei stehen. Sie definieren die xml-Version und laden die T_EXnicCenter-spezifische xsd-Datei. Die Befehle `<lxPackage name="mathematik">` bzw. `<lxPackage name="griechische Buchstaben">` und `</lxPackage>` (Zeile 3 und 15 bzw. 17 und 25) ermöglichen eine Gliederung der xml-Datei nach Paketnamen bzw. Anwendungszweck der Befehle. Innerhalb dieser Befehle stehen die eigentlichen Autovervollständigingsdefinitionen: `<lxEnvironment name="name" ... />` Diese werden im Folgenden anhand der Definitionen der Umgebung `equation*` sowie der Befehle `\frac` und `\alpha` genauer betrachtet.

Definition von `equation*`

Zuerst wird der Name des Befehls mittels `name="equation"` vergeben. Dies ist die Zeichenfolge, welche später in T_EXnicCenter eingetippt werden muss. Die folgende Zeile `parameters="0"` definiert die Anzahl der Parameter, die ein Befehl hat. Offenbar hat dieser Wert aber keinen Einfluss auf die Autovervollständigung. Ich lasse ihn deshalb stets auf 0. `desc="Mathebereich"` definiert den Hilfetext, welcher im Autovervollständigungsfenster rechts neben dem Befehl angezeigt wird. `expafter="}
	
\end{equation}"` bestimmt, was nach dem Befehl (in diesem Fall `equation*`) eingefügt werden soll: Eine schließende geschweifte Klammer `}`, gefolgt von einem Zeilenvorschub (`
`), einem Tabulatoreinzug (`	`), einem weiteren Zeilenvorschub (`
`) und `\end{equation}`. Der Befehl `expbefore="\begin{"` bestimmt, was vor dem Befehl eingefügt werden soll, nämlich ein `\begin{`.

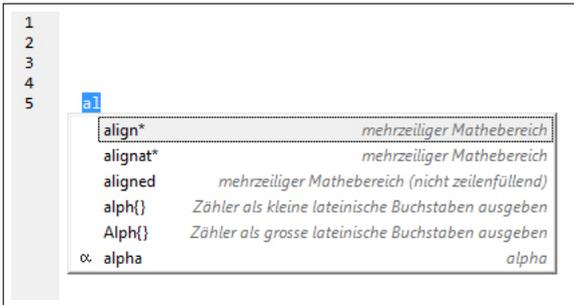
Definition von `\frac`

Die Definition von `\frac` unterscheidet sich nur leicht von der `equation*`-Definition. Der Name ist hier `frac{ }{ }`. Zwar könnten die beiden Klammerpaare auch mittels `expafter="{}{ }` definiert werden. Mit der ersten Methode springt T_EXnicCenter nach der Autovervollständigung aber direkt ins erste Klammerpaar. Nach `frac` wird also nichts ausgegeben (`expafter=""`), davor wird mittels `expbefore="\` der Backslash `\` gesetzt.

Definition von `\alpha`

Bei der Definition von `\alpha` gibt es zwei besondere Befehle, `icon="symbols.bmp"` und `index="0"`. Mit diesen Befehlen kann ein Icon, welches im Autovervoll-

ständigungsfenster links vom Befehl erscheint (siehe folgenden Screenshot), definiert werden. Einige Symboldateien (`accents.bmp`, `arrows.bmp`, `binops.bmp`, `format.bmp`, `mathmisc.bmp`, `sets.bmp`, `symbols.bmp` und `symbols2.bmp`) werden mit \TeX nicCenter mitgeliefert. Neue können mit jedem Grafikprogramm erstellt werden.



`icon="symbols.bmp"` bestimmt die Datei, aus der das Symbol entnommen werden soll. `index="0"` bestimmt die Position. `index="0"` heißt, das erste Symbol wird verwendet, `index="1"` heißt, das zweite Symbol wird verwendet, usw.

Bemerkung zu Steuerzeichen

Steuerzeichen (Tabulator, Zeilenvorschub, Wagenrücklauf und Leerzeichen) werden in den Autovervollständigungsdateien mit Hilfe der Hexadezimalwerte der entsprechenden Unicode-Zeichen eingegeben.

Tabulator	Zeilenvorschub	Wagenrücklauf	Leerzeichen
<code>&#x9;</code>	<code>&#xA;</code>	<code>&#xD;</code>	<code>&#x20;</code>

Der Tabulator und das Leerzeichen können (zumindest in meinem Editor Notepad++) ohne Verwendung der entsprechenden Hexadezimalwerte – also durch Drücken der entsprechenden Tasten auf der Tastatur – eingegeben werden. Der Zeilenvorschub und der Wagenrücklauf sind rein optisch nicht zu unterscheiden; ich verwende stets den Zeilenvorschub – entsprechend dem Vorbild der mit \TeX nicCenter mitgelieferten Autovervollständigungsdateien.

Tipps und Tricks

Das Paket `minted` und der Apostroph

Marco Daniel

Das Hervorheben von Programmcode in \LaTeX mit Hilfe von `Pygmentize` (*Python syntax highlighter*) [1] wird immer beliebter. Ein kleiner Nachteil dieser Methode sind gerade Apostrophe, welche in diversen Programmiersprachen häufig benötigt werden. Diese Apostrophe werden in der Ausgabe mit Serifen dargestellt. Der Artikel zeigt, wie dieses Manko behoben werden kann.

Einleitung

Die Darstellung von Codesequenzen in \LaTeX -Dateien ist integraler Bestandteil vieler Arbeiten. Hierbei ist die Bestrebung nach farblichen Hervorhebungen, wie es die meisten Editoren können, ein primäres Ziel. Die bekanntesten Pakete hierzu sind wohl `listings` [3] und `fancyvrb` [6]. Neuere Vertreter sind die Pakete `minted` [4] und `verbments` [5], welche zum Hervorheben `Pygmentize` [1] und zur Ausgabe `fancyvrb` [6] nutzen.

Die Darstellung bei der Eingabe von geraden Apostrophen führt bei den neueren Vertretern unter Nutzung von `pdf \LaTeX` zur Ausgabe eines geschwungenen Apostrophs. Vermutlich kommt Ihnen das Paket `upquote` in den Sinn, was in diesem Fall keinen Erfolg bringt.

Der Artikel soll zeigen, wie dieser Nachteil behoben werden kann.

Apostrophe unter Nutzung von `Lua \LaTeX` oder `X \LaTeX`

In der Einleitung ist die Kompilierermethode `pdf \LaTeX` erwähnt worden. Bei der Anwendung von `Lua \LaTeX` oder `X \LaTeX` wird das gewünschte Ergebnis ohne weitere Maßnahmen erreicht. Folgendes Beispiel soll dies verdeutlichen.

```
% !TEX program = lualatex --shell-escape
\documentclass{article}
\usepackage{fontspec}
\usepackage{minted}
```

```
\begin{document}
\begin{minted}{bash}
grep 'foo' *.txt
\end{minted}
\end{document}
```

```
grep 'foo' *.txt
```

Abbildung 1: Ausgabe mit `fontspec` und `lualatex`.

Apostrophe unter Nutzung von pdf_lAT_EX

Das Problem der Apostrophe ist schon viele Jahre bekannt und bereits im Jahr 2000 versuchte das Paket `upquote` [2] dieses Problem zu beheben. Bis zur Veröffentlichung von `Pygmentize` Version 1.6 funktionierte `upquote` ebenfalls in Zusammenarbeit mit `minted` [4] und `verbments` [5]. So genügte es, das Paket zu laden, um die gewünschte Ausgabe zu erhalten.

```
% !TEX program = pdflatex --shell-escape
\documentclass{article}
\usepackage[T1]{fontenc}
\usepackage{minted}
\usepackage{upquote}
\begin{document}
\begin{minted}{bash}
grep 'foo' *.txt
\end{minted}
\end{document}
```

Mit der Version 1.6 hat `Pygmentize` den $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ -Lexer erweitert, so dass die bisherige Methode leider erfolglos ist, wie die Abbildung 2 verdeutlicht.

```
grep 'foo' *.txt
```

Abbildung 2: Geschwungene Apostrophe, obwohl das Paket `upquote` geladen wurde.

Die Version von `Pygmentize` kann über das Terminal durch die nachstehende Eingabe ermittelt werden.

```
marco@imac:~$pygmentize -V
Pygments version 1.6, (c) 2006–2013 by Georg Brandl.
```

Um dennoch das gewünschte Ergebnis zu erzielen, genügt es, folgende Zeile in der Präambel zu ergänzen.

```
\AtBeginDocument{\let\PYZsq\textquotesingle}
```

Wird diese Deklaration zum obigen Beispiel hinzugefügt, so erhalten wir die gewünschte Ausgabe. Die nachstehende Abbildung soll das Ergebnis noch einmal veranschaulichen.

Abbildung 3: Gerade Apostrophe, wie man sie erwartet.

```
grep 'foo' *.txt
```

Literatur

- [1] Georg Brandl, Tim Hatch und Armin Ronacher: *Pygments – python syntax highlighter*; 2013; <http://pygments.org>.
- [2] Michael A. Covington, Frank Mittelbach und Markus G. Kuhn: *upquote – upright-quote and grave-accent glyphs in verbatim*; version 1.3; 2012; CTAN:macros/latex/contrib/upquote/.
- [3] Carsten Heinz und Brooks Moses: *The Listings package; version 1.4*; 2011; CTAN:macros/latex/contrib/listings/.
- [4] Konrad Rudolph: *The minted package: Highlighted source code in L^AT_EX*; version 1.7; 2011; CTAN:macros/latex/contrib/minted/.
- [5] Dejan Živković: *The verbments package: Pretty printing source code in L^AT_EX*; version 1.2; 2011; CTAN:macros/latex/contrib/verbments/.
- [6] Timothy Van Zandt: *The fancyvrb package – Fancy Verbatims in L^AT_EX*; version 2.8; 2011; CTAN:macros/latex/contrib/fancyvrb/.

Eulersche Gerade mit tkz-euclide zeichnen

Roger Jud

Anhand der Konstruktion der eulerschen Geraden werden einige Funktionen des Pakets tkz-euclide gezeigt.

tkz-euclide

Das Paket tkz-euclide von Alain Matthes baut auf pgf/TikZ von Till Tantau auf und erlaubt die Erstellung geometrischer Zeichnungen mittels entsprechender Befehle direkt im L^AT_EX-Code. Geladen wird das Paket wie üblich in der Präambel mit folgenden Befehlen:

```
\usepackage{tkz-euclide}
\usetkzobj{all}
```

Der Befehl `\usetkzobj{all}` stellt sicher, dass alle tkz-Befehle verfügbar sind. Würde man beispielsweise nur Kreis-Befehle benötigen, könnte man mittels `\usetkzobj{circles}` auch nur diese Befehle laden. Da aber meist ein großer Teil der Befehle benötigt wird, scheint `\usetkzobj{all}` die bessere Variante zu sein. Die eigentlichen Zeichnungen werden nach `\begin{document}` innerhalb von `tikzpicture`-Umgebungen erstellt.

```
% Präambel mit Angabe der Dokumentklasse
% weitere Pakete laden
\usepackage{tkz-euclide}
\usetkzobj{all}
\begin{document}
\begin{tikzpicture}
% Zeichnungsbefehle
\end{tikzpicture}
\end{document}
```

Schrittweises Erstellen der eulerschen Geraden

Die eulersche Gerade ist nach dem Schweizer Mathematiker Leonard Euler benannt, welcher im 18. Jahrhundert lebte. Den meisten (zumindest Schweizern) ist er von der alten Schweizer 10-Franken-Note her bekannt.

Unter der eulerschen Geraden eines Dreiecks versteht man die Gerade, die durch den Schwerpunkt S , den Umkreismittelpunkt U und den Höhenschnittpunkt H des Dreiecks geht. Da eine Gerade durch zwei Punkte eindeutig definiert ist, genügen zur Konstruktion der eulerschen Geraden zwei der obigen Punkte. Wir konstruieren hier deshalb nur S und U .

Der Zeichnungsbereich

Zuerst benötigen wir einen Zeichnungsbereich:

```
\begin{tikzpicture}
\tkzInit[xmin=-1,xmax=7,ymin=-1,ymax=4]
\tkzClip
\end{tikzpicture}
```

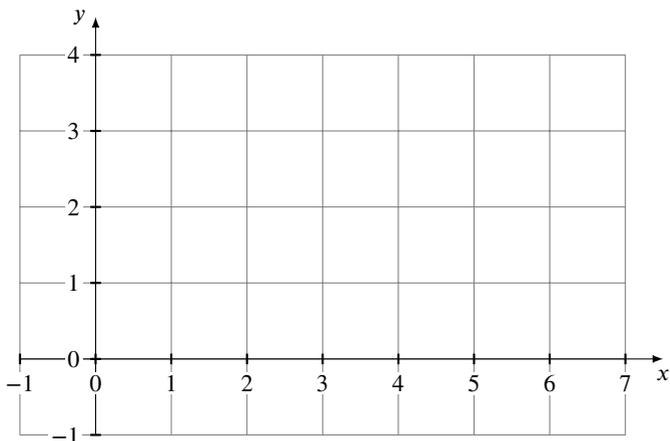
`\tkzInit[xmin=-1,xmax=7,ymin=-1,ymax=4]` definiert den Bereich, in dem gezeichnet werden soll. `\tkzClip` schneidet diesen Bereich zu. Sollte also irgendein Objekt aus dem definierten Bereich hinausragen, so wird es abgeschnitten. Wer Mühe hat, sich im Koordinatensystem zurechtzufinden, kann mit `\tkzGrid` ein Raster und mit `\tkzAxeXY` x - und y -Achsen zeichnen. Werden Raster und Achsen nicht mehr benötigt, können sie auskommentiert werden.

```
\begin{tikzpicture}
\tkzInit[xmin=-1,xmax=7,ymin=-1,ymax=4]
```

```

\tkzGrid
\tkzAxeXY
\tkzClip
\end{tikzpicture}

```



Der Befehl `\tkzClip` sollte stets nach `\tkzAxeXY` stehen, da sonst die Pfeile sowie die Beschriftungen der Achsen abgeschnitten werden.

Das Dreieck

Mit dem Paket `tkz-euclide` werden sämtliche Objekte mit Hilfe von Punkten definiert. Entsprechend benötigen wir drei Punkte A , B und C , die wir mit der Anweisung `\tkzDefPoints{0/0/A, 6/0/B, 4.5/3/C}` definieren, mittels `\tkzDrawPolygon(A,B,C,A)` verbinden, mittels `\tkzDrawPoints(A,B,C)` zeichnen und schließlich mit Hilfe des Befehls `\tkzLabelPoint` beschriften.

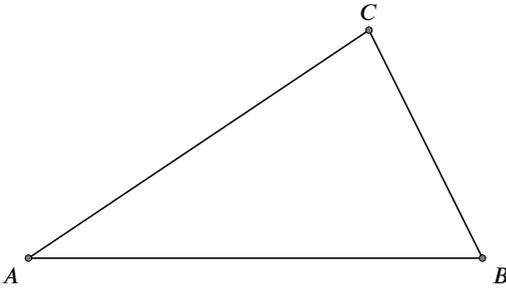
```

\begin{tikzpicture}
\tkzInit[xmin=-1,xmax=7,ymin=-1,ymax=4]
\tkzClip
\tkzDefPoints{0/0/A, 6/0/B, 4.5/3/C}

\tkzDrawPolygon(A,B,C,A)
\tkzDrawPoints(A,B,C)

\tkzLabelPoint[below left](A){$A$}
\tkzLabelPoint[below right](B){$B$}
\tkzLabelPoint[above](C){$C$}
\end{tikzpicture}

```



Der Schwerpunkt

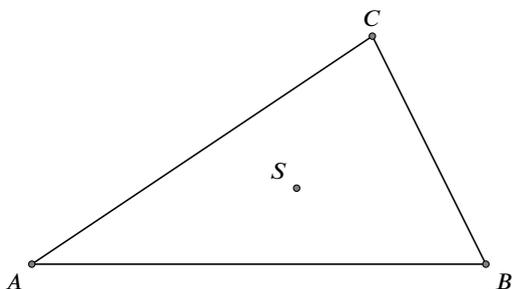
Der Schwerpunkt ist der Schnittpunkt der drei Schwerlinien (auch Seitenhalbierende genannt), welche vom Eckpunkt eines Dreiecks zum gegenüberliegenden Seitenmittelpunkt gehen. Es genügt die Konstruktion von zwei Schwerlinien, da deren Schnittpunkt den Schwerpunkt eindeutig definiert. Mit Hilfe des Befehls `\tkzDefMidPoint(A,B) \tkzGetPoint{MAB}` bestimmen wir den Mittelpunk $M_{\overline{AB}}$ der Dreiecksseite \overline{AB} , `\tkzDefMidPoint(B,C) \tkzGetPoint{MBC}` gibt den Mittelpunk $M_{\overline{BC}}$ der Dreiecksseite \overline{BC} . Mittels `\tkzInterLL(C,MAB)(A,MBC) \tkzGetPoint{S}` lassen wir die zwei Schwerlinien schneiden und erhalten so den Schwerpunkt S .

```
\begin{tikzpicture}
  \tkzInit[xmin=-1,xmax=7,ymin=-1,ymax=4]
  \tkzClip
  \tkzDefPoints{0/0/A, 6/0/B, 4.5/3/C}

  \tkzDefMidPoint(A,B) \tkzGetPoint{MAB}
  \tkzDefMidPoint(B,C) \tkzGetPoint{MBC}
  \tkzInterLL(C,MAB)(A,MBC) \tkzGetPoint{S}

  \tkzDrawPolygon(A,B,C,A)
  \tkzDrawPoints(A,B,C,S)

  \tkzLabelPoint[below left](A){$A$}
  \tkzLabelPoint[below right](B){$B$}
  \tkzLabelPoint[above](C){$C$}
  \tkzLabelPoint[above left](S){$S$}
\end{tikzpicture}
```



Der Umkreismittelpunkt

Der Umkreismittelpunkt ist der Schnittpunkt der Mittelsenkrechten der drei Seiten. Auch hier genügt die Konstruktion zweier Mittelsenkrechten, da deren Schnittpunkt den Umkreismittelpunkt eindeutig definiert. Die beiden Befehle `\tkzDefLine[orthogonal=through MAB](A,B)` `\tkzGetPoint{mab}` definieren einen Punkt `mab`, der auf der Senkrechten zur Seite \overline{AB} durch den Seitenmittelpunkt $M_{\overline{AB}}$ liegt. Entsprechendes gilt für die Seite \overline{BC} . Die beiden Mittelsenkrechten werden wiederum mit dem Befehl `\tkzInterLL(MAB,mab)(MBC,mbc)` `\tkzGetPoint{U}` geschnitten; der Umkreismittelpunkt U ist somit definiert.

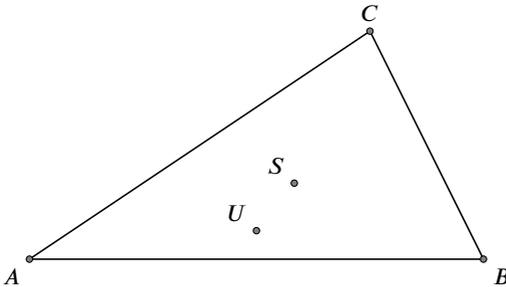
```
\begin{tikzpicture}
  \tkzInit[xmin=-1,xmax=7,ymin=-1,ymax=4]
  \tkzClip
  \tkzDefPoints{0/0/A, 6/0/B, 4.5/3/C}

  \tkzDefMidPoint(A,B) \tkzGetPoint{MAB}
  \tkzDefMidPoint(B,C) \tkzGetPoint{MBC}
  \tkzInterLL(C,MAB)(A,MBC) \tkzGetPoint{S}

  \tkzDefLine[orthogonal=through MAB](A,B) \tkzGetPoint{mab}
  \tkzDefLine[orthogonal=through MBC](B,C) \tkzGetPoint{mbc}
  \tkzInterLL(MAB,mab)(MBC,mbc) \tkzGetPoint{U}

  \tkzDrawPolygon(A,B,C,A)
  \tkzDrawPoints(A,B,C,S,U)

  \tkzLabelPoint[below left](A){$A$}
  \tkzLabelPoint[below right](B){$B$}
  \tkzLabelPoint[above](C){$C$}
  \tkzLabelPoint[above left](S){$S$}
  \tkzLabelPoint[above left](U){$U$}
\end{tikzpicture}
```



Die eulersche Gerade

`\tkzDrawLine[add = 1.5 and 3.0, end=e](U,S)` zeichnet schließlich die eulersche Gerade e . Damit die Gerade über U bzw. S hinausgeht, verlängern wir sie mittels `add = 1.5` and `3.0` auf der Seite von U um 150% und auf der Seite von S um 300% der Länge der Strecke \overline{US} . `end=e` setzt die Beschriftung ans Ende der Geraden.

```
\begin{tikzpicture}
  \tkzInit[xmin=-1,xmax=7,ymin=-1,ymax=4]
  \tkzClip
  \tkzDefPoints{0/0/A, 6/0/B, 4.5/3/C}

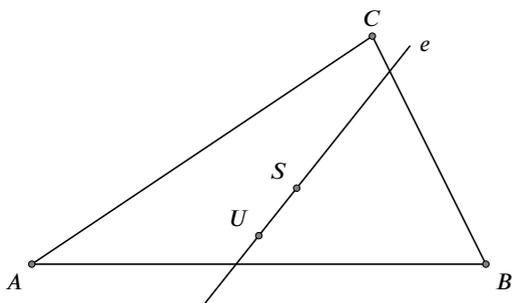
  \tkzDefMidPoint(A,B) \tkzGetPoint{MAB}
  \tkzDefMidPoint(B,C) \tkzGetPoint{MBC}
  \tkzInterLL(C,MAB)(A,MBC) \tkzGetPoint{S}

  \tkzDefLine[orthogonal=through MAB](A,B) \tkzGetPoint{mab}
  \tkzDefLine[orthogonal=through MBC](B,C) \tkzGetPoint{mbc}
  \tkzInterLL(MAB,mab)(MBC,mbc) \tkzGetPoint{U}

  \tkzDrawPolygon(A,B,C,A)
  \tkzDrawLine[add = 1.5 and 3.0, end=$e$](U,S)

  \tkzDrawPoints(A,B,C,S,U)

  \tkzLabelPoint[below left](A){$A$}
  \tkzLabelPoint[below right](B){$B$}
  \tkzLabelPoint[above](C){$C$}
  \tkzLabelPoint[above left](S){$S$}
  \tkzLabelPoint[above left](U){$U$}
\end{tikzpicture}
```



Struktur einer Konstruktion

Es hat sich in der Praxis bewährt, Konstruktionen wie folgt zu strukturieren:

1. Definition und Zuschneiden des Zeichnungsbereichs mittels `\tkzInit` und `\tkzClip`.
2. Zeichnen des Rasters mittels `\tkzGrid`. (Dieses dient lediglich als Orientierungshilfe im Koordinatensystem und wird später auskommentiert oder entfernt.)
3. Definition sämtlicher Fixpunkte mittels `\tkzDefPoint` bzw. `\tkzDefPoints`.
4. Konstruktion abhängiger Punkte mittels `\tkzInterLL`, `\tkzDefMidPoint`, usw.
5. Zeichnen von Winkelbögen, Kreisen und Kreisbögen.
6. Zeichnen von rechten Winkeln.
7. Zeichnen von Geraden, Strecken und Polygonen.
8. Zeichnen von Parallelzeichen.
9. Zeichnen von Punkten.
10. Beschriften sämtlicher Objekte.

Schleifenmakro

Herbert Voß

Es gibt verschiedene Pakete, die Makros für Schleifen anbieten. Da mittlerweile alle \TeX -Programme auf der Basis von ϵ - \TeX arbeiten, kann man sich auch ein eigenes Schleifenmakro schreiben, welches mit einer Rekursion arbeitet. [1]

```
\def\replicate#1#2{\ifnum#1>0 #2%
\expandafter\replicate\expandafter{\number\numexpr#1-1}\fi}
```

Der Parameter #1 definiert die Anzahl der Wiederholungen und #2 das zu wiederholende Objekt.

Ist \ifnum\#1>0 eine wahre Aussage, so wird zum einen das angegebene Objekt #2 ausgeführt und zum anderen durch die beiden \expandafter dafür gesorgt, dass vor dem erneuten Aufruf des Makros \replicate (Rekursion) $\text{\number\numexpr\#1-1}$ expandiert wird. Es wird also eine neue Zahl als Parameter #1 erzeugt, die gegenüber der vorigen um 1 vermindert ist.

```
\newlength\Abstand
\replicate{5}{%
\hspace*{\Abstand}\TeX
\addtolength\Abstand{1cm}%
\newline}
```

\TeX

\TeX

\TeX

\TeX

\TeX

Literatur

- [1] David Kastrup: *Itarting with roman numerals*; 2005; <http://www.gust.org.pl/projects/pearls/2005p/david-kastrup/bachotex2005-david-kastrup-pearl3.pdf>; zuletzt besucht: 2013-07-31.

	ConT _E Xt	LuaL ^A T _E X	X _Q L ^A T _E X	MS Word 2010
Latin Modern Math	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$
TG Bonum Math	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$
TG Pagella Math	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$
TG Termes Math	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$
Cambria Math	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$
Asana Math	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$
Lucida Math	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$
XITS Math	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedm}}$ $\overbrace{aaaaa}^{\text{pi}}$	$\overbrace{aaaaaaaa}^{\text{Siedem}}$ $\overbrace{aaaaa}^{\text{piec}}$

Latin Modern Math

$$\frac{\partial \mathbf{H}_{i\alpha, i\beta}}{\partial \bar{R}_k} = \frac{\partial \mathbf{H}_{i\alpha, i\beta}}{\partial \varrho_i} \frac{\partial \varrho_i}{\partial \bar{R}_k} = \frac{2}{3} (b_q \varrho_i^{-1/3} + 2c_q \varrho_i^{1/3}) \frac{\partial}{\partial \bar{R}_k} \left(\sum_{j \neq i} e^{(-\lambda^2 R_{ij})} F_c(R_{ij}) \right)$$

PDFL^AT_EX (CM fonts)

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

ConT_EXt

$$\frac{\partial \mathbf{H}_{i\alpha, i\beta}}{\partial \bar{R}_k} = \frac{\partial \mathbf{H}_{i\alpha, i\beta}}{\partial \varrho_i} \frac{\partial \varrho_i}{\partial \bar{R}_k} = \frac{2}{3} (b_q \varrho_i^{-1/3} + 2c_q \varrho_i^{1/3}) \frac{\partial}{\partial \bar{R}_k} \left(\sum_{j \neq i} e^{(-\lambda^2 R_{ij})} F_c(R_{ij}) \right)$$

LuaL^AT_EX

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

X_QL^AT_EX

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

\TeX Gyre Bonum Math

$$\frac{\partial \mathbf{H}_{ia, i\epsilon}}{\partial \bar{R}_k} = \frac{\partial \mathbf{H}_{ia, i\epsilon}}{\partial \rho_i} \frac{\partial \rho_i}{\partial \bar{R}_k} = \frac{2}{3} (b_q \rho_i^{-1/3} + 2c_q \rho_i^{1/3}) \frac{\partial}{\partial \bar{R}_k} \left(\sum_{j \neq i} e^{(-\beta^2 R_{ij})} F_c(R_{ij}) \right)$$

Con \TeX t

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

$$\frac{\partial \mathbf{H}_{ia, i\epsilon}}{\partial \bar{R}_k} = \frac{\partial \mathbf{H}_{ia, i\epsilon}}{\partial \rho_i} \frac{\partial \rho_i}{\partial \bar{R}_k} = \frac{2}{3} (b_q \rho_i^{-1/3} + 2c_q \rho_i^{1/3}) \frac{\partial}{\partial \bar{R}_k} \left(\sum_{j \neq i} e^{(-\beta^2 R_{ij})} F_c(R_{ij}) \right)$$

Lua \LaTeX

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

$$\frac{\partial \mathbf{H}_{ia, i\epsilon}}{\partial \bar{R}_k} = \frac{\partial \mathbf{H}_{ia, i\epsilon}}{\partial \rho_i} \frac{\partial \rho_i}{\partial \bar{R}_k} = \frac{2}{3} (b_q \rho_i^{-1/3} + 2c_q \rho_i^{1/3}) \frac{\partial}{\partial \bar{R}_k} \left(\sum_{j \neq i} e^{(-\beta^2 R_{ij})} F_c(R_{ij}) \right)$$

X \LaTeX

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

\TeX Gyre Pagella Math

$$\frac{\partial \mathbf{H}_{ia, i\beta}}{\partial \bar{R}_k} = \frac{\partial \mathbf{H}_{ia, i\beta}}{\partial \rho_i} \frac{\partial \rho_i}{\partial \bar{R}_k} = \frac{2}{3} (b_q \rho_i^{-1/3} + 2c_q \rho_i^{1/3}) \frac{\partial}{\partial \bar{R}_k} \left(\sum_{j \neq i} e^{(-\lambda^2 R_{ij})} F_c(R_{ij}) \right)$$

Con \TeX t

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

$$\frac{\partial \mathbf{H}_{ia, i\beta}}{\partial \bar{R}_k} = \frac{\partial \mathbf{H}_{ia, i\beta}}{\partial \rho_i} \frac{\partial \rho_i}{\partial \bar{R}_k} = \frac{2}{3} (b_q \rho_i^{-1/3} + 2c_q \rho_i^{1/3}) \frac{\partial}{\partial \bar{R}_k} \left(\sum_{j \neq i} e^{(-\lambda^2 R_{ij})} F_c(R_{ij}) \right)$$

Lua \LaTeX

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

$$\frac{\partial \mathbf{H}_{ia, i\beta}}{\partial \bar{R}_k} = \frac{\partial \mathbf{H}_{ia, i\beta}}{\partial \rho_i} \frac{\partial \rho_i}{\partial \bar{R}_k} = \frac{2}{3} (b_q \rho_i^{-1/3} + 2c_q \rho_i^{1/3}) \frac{\partial}{\partial \bar{R}_k} \left(\sum_{j \neq i} e^{(-\lambda^2 R_{ij})} F_c(R_{ij}) \right)$$

X \LaTeX

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

T_EX Gyre Termes Math

$$\frac{\partial \mathbf{H}_{i\alpha,i\beta}}{\partial \vec{R}_k} = \frac{\partial \mathbf{H}_{i\alpha,i\beta}}{\partial \varrho_i} \frac{\partial \varrho_i}{\partial \vec{R}_k} = \frac{2}{3} (b_q \varrho_i^{-1/3} + 2c_q \varrho_i^{1/3}) \frac{\partial}{\partial \vec{R}_k} \left(\sum_{j \neq i} e^{(-\lambda^2 R_{ij})} F_c(R_{ij}) \right)$$

ConT_EXt

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

$$\frac{\partial \mathbf{H}_{i\alpha,i\beta}}{\partial \vec{R}_k} = \frac{\partial \mathbf{H}_{i\alpha,i\beta}}{\partial \varrho_i} \frac{\partial \varrho_i}{\partial \vec{R}_k} = \frac{2}{3} (b_q \varrho_i^{-1/3} + 2c_q \varrho_i^{1/3}) \frac{\partial}{\partial \vec{R}_k} \left(\sum_{j \neq i} e^{(-\lambda^2 R_{ij})} F_c(R_{ij}) \right)$$

LuaL^AT_EX

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

$$\frac{\partial \mathbf{H}_{i\alpha,i\beta}}{\partial \vec{R}_k} = \frac{\partial \mathbf{H}_{i\alpha,i\beta}}{\partial \varrho_i} \frac{\partial \varrho_i}{\partial \vec{R}_k} = \frac{2}{3} (b_q \varrho_i^{-1/3} + 2c_q \varrho_i^{1/3}) \frac{\partial}{\partial \vec{R}_k} \left(\sum_{j \neq i} e^{(-\lambda^2 R_{ij})} F_c(R_{ij}) \right)$$

X_GL^AT_EX

$$\frac{\ln \left(\lim_{z \rightarrow \infty} \left(\left((\bar{X}^T)^{-1} - (\bar{X}^{-1})^T \right) + \frac{1}{z} \right)^2 \right) + \sin^2(p) + \cos^2(p)}{\sum_{n=0}^{\infty} \frac{\cosh(q) \cdot \sqrt{1 - \tanh^2(q)}}{2^n}} = \frac{1}{2} \sum_{n=0}^{\infty} \left(\frac{128}{2^8} \right)^n$$

Von fremden Bühnen

Neue Pakete auf CTAN

Jürgen Fenn

Der Beitrag stellt neue Pakete auf CTAN seit der letzten Ausgabe bis zum Redaktionsschluss vor. Bloße Updates können auf der moderierten *CTAN-ann*-Mailingliste verfolgt werden, die auch auf Twitter als @ctanannounce verfügbar ist.

gentium-tug von *Karl Berry* enthält die Schriftfamilie Gentium für den Satz lateinischer, kyrillischer und griechischer Schriften in den Formaten TrueType und Type 1 mit der entsprechenden Unterstützung für \LaTeX und \ConTeXt .

CTAN:fonts/gentium-tug

uowthesistitlepage von *Thomas M. Griffiths* dient zum Setzen der Titelseite einer Abschlussarbeit der *University of Wollongong*.

CTAN:macros/latex/contrib/uowthesistitlepage

changes von *Ekkart Kleinod* ermöglicht es, Änderungen und Streichungen in einem Text händisch auszuzeichnen und entsprechend zu setzen, einschließlich einer Liste aller Änderungen.

CTAN:macros/latex/contrib/changes

lplfitch von *Richard Zach* dient zum Setzen natürlicher Schlüsse im Stil von Fitch, wie sie in dem Lehrbuch »Language, Proof, and Logic« von Dave Barker-Plummer, Jon Barwise und John Etchemendy verwendet werden (daher der Name des Pakets).

CTAN:macros/latex/contrib/lplfitch

expex von *John Frampton* ist eine Sammlung von Makros für das Setzen von Beispielen und Glossen in linguistischen Arbeiten.

CTAN:macros/plain/contrib/expex

librecaslon von *Bob Tennent* enthält den Font *Libre Caslon* nebst \LaTeX -Unterstützung.

CTAN:fonts/librecaslon

udesoftec von *J. Peter* und *M. Schuler* enthält Vorlagen für »Qualifikationsarbeiten« am Lehrstuhl für Wirtschaftsinformatik und Softwaretechnik der Universität Duisburg-Essen.

CTAN:macros/latex/contrib/udesoftec

create-struktex von *Peter Lenser* dient dazu, eine \TeX -Datei aus Java-Code zu erzeugen, um Nassi-Shneiderman-Diagramme mit dem Paket *struktex* zu erzeugen.

CTAN:support/create-struktex

stackengine von *Steven B. Segletes* erlaubt es, im Textmodus Objekte in »Stacks« übereinander zu setzen.

CTAN:macros/latex/contrib/stackengine

latex-git-log von *Robin Schneider* ist ein Programm, das die gesamte Versionsgeschichte eines Git-Projekts in Form einer \LaTeX -Tabelle ausgibt, wenn es innerhalb eines Git-Repositorys ausgeführt wird.

CTAN:support/latex-git-log

feuphdteses von *Mario S. E.* ist eine Vorlage für eine Doktorarbeit an der ingenieurwissenschaftlichen Fakultät der *Universidade do Porto (FEUP)*.

CTAN:macros/latex/contrib/feuphdteses

nox von *Matthias Borck-Elsner* dient zum Umsetzen von Arrays, Texten, Befehlen und Umgebungen in »wohlstrukturierte und teilbare Tabellen«.

CTAN:macros/latex/contrib/nox

thalie von *Louis Paternault* ist ein weiteres \LaTeX -Paket zum Setzen von Theaterstücken.

CTAN:macros/latex/contrib/thalie

classpack von *Peter Flynn* enthält eine »funktionierende, aber unvollständige Version« des XML-Mastering-Systems, das der Autor auf der TUG 2012 in Boston vorgestellt hatte.

CTAN:support/classpack

luabidi von *Arthur Reutenauer* ist eine neue Fassung des Pakets *bidi* von *Vafa Khalighi* für \LaTeX . Die Dokumentation folgt.

CTAN:macros/luatex/latex/luabidi

textglos von *Natalie Weber* dient zum Setzen in Beispielen in linguistischen Texten (inline).

CTAN:macros/latex/contrib/textglos

leipzig von *Natalie Weber* dient zum Setzen und Indizieren von Beispiel-Abkürzungen in linguistischen Texten.

CTAN:macros/latex/contrib/leipzig

interval von *Lars Madsen* sorgt beim Setzen von Intervallen für das korrekte Spacing.

CTAN:macros/latex/contrib/interval

selnolig von *Mico Loretan* entfernt unpassende Ligaturen in deutschen und englischen Texten (benötigt \LaTeX).

CTAN:macros/luatex/latex/selnolig

latexindent von *Chris Hughes* ist ein Perl-Skript für den Einzug in \LaTeX -Dokumenten.

CTAN:support/latexindent

conteg von *Joachim Breitner* erlaubt es, eine über mehrere Zeilen gehende Gleichung in mehreren verschiedenen Layouts zu setzen.

CTAN:macros/latex/contrib/conteg

detexfaq von *Patrick Gundlach* bringt die neugefasste deutschsprachige TeX-FAQ im PDF-Format auf CTAN. Das Projekt ist auf <http://texfragen.de> beheimatet und wird dort unter der Lizenz CC-by-sa 3.0 weiter bearbeitet. Jeder kann mitmachen.

CTAN:info/german/detexfaq

babel-thai von *Theppitak Karoonboonyanan* enthält die Unterstützung für babel aus dem Thai \TeX -Projekt. Das Paket *thai*latex wird damit obsolet.

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

horoscop von *Matthew Skala* erlaubt es, astrologische Grafiken mit *pict2e* zu setzen. Berechnungen können aus den Anwendungen *Astrolog* und *Swiss Ephemeris* übernommen werden.

CTAN:macros/latex/contrib/horoscop

greek-fontenc von *Günter Milde* stellt die *font encoding definition files* für griechische Text-fonts bereit.

CTAN:language/greek/greek-fontenc

wheretotrim von *Scott Pakin* besteht aus einem Perl-Skript und einem ergänzenden \TeX -Paket, das dabei helfen soll, Text gleichmäßig innerhalb eines Dokuments zu verteilen, um die insgesamt benötigte Seitenzahl zu optimieren. Dazu werden Kürzungsvorschläge zum Trimmen des Texts gemacht (nur unter Linux getestet).

CTAN:support/wheretotrim

memory von *Song Zhiwei* ermöglicht es, Variablen als Array und Objekte wie in anderen Programmiersprachen zu deklarieren.

CTAN:macros/latex/contrib/memory

try von *Ajabu Tex* ist ein weiteres Programm, mit dem man die \TeX -Läufe automatisieren kann.

CTAN:support/try

raleway von *Silke Hofstra* enthält die \TeX -Unterstützung für die Schrift *Raleway*. Mit \XeTeX und \LuaTeX werden OpenType-Fonts verwendet.

CTAN:fonts/raleway

parsitex von der *Persian TeX Group* ist eine lokalisierte und bidirektionale Erweiterung von \TeX , die auf dem \TeX - \XeTeX -Algorithmus von Donald E. Knuth beruht, aber die Beschränkungen von *etex* behebt.

CTAN:systems/parsitex

bxdpx-beamer von *Takayuki Yato* behebt einige Probleme, die bei der Verwendung des Pakets *beamer* mit *dvipdfmx* auftreten.

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

pxpgfmark von *Takayuki Yato* erlaubt es, *inner-picture connections* mit *PGF* und *dvipdfmx* zu verwenden.

CTAN:graphics/pgf/contrib/pxpgfmark

factura von *Emilio Lazo Zaia* braucht jeder, der für Venezuela spezifische Rechnungen ausstellen möchte.

CTAN:macros/latex/contrib/factura

sansmathfonts von *Ariel Barton* stellt serifenlose Kapitälchen und Mathematikfont für Computer Modern bereit.

CTAN:fonts/sansmathfonts

pythontex von *Geoffrey Poore* ermöglicht es, Python-Code in einem \TeX -Dokument einzugeben, auszuführen und das Ergebnis in den Output einfließen zu lassen.

CTAN:macros/latex/contrib/pythontex

esami von *Grazia Messineo* und *Salvatore Vassallo* dient zum Setzen von Prüfungs- und Fragebögen. Fragen und Antworten können dabei zufällig und in mehreren Sprachen ausgegeben werden.

CTAN:macros/latex/contrib/esami

Im Netz gefunden

Herbert Voß

In den verschiedenen Mailinglisten, Webforen, Newsgroups, u. a., findet man immer wieder hilfreiche Angaben zur Arbeit mit und um das Thema Textsatz mit \TeX , \LaTeX , Con \TeX t, usw.

Schreibweise des Plurals eines Initialwortes

Auf unserer vereinseigenen Mailingliste findet sich eine Anmerkung von Peter Gallmann zur Schreibweise des Plurals eines Initialwortes, beispielsweise von »URL«. Ausgangspunkt war die Diskussion, wie fremdsprachliche Termini zu übersetzen sind und ob sie der naheliegenden Übersetzung entsprechen sollten.

Für Akronyme gilt die Regel nicht: Entweder s-Plural oder endunglos. Duden-Grammatik (2009), Randziffer 290. Also *weder* URLEN noch URLS, sondern URLs (mit kleinem s) oder auch nur URL.

Spielplan

Termine

2013

23. 10. – 26. 10. **TUG 2013**
Tokio, Japan
<http://tug.org/tug2013/>
2. 11. **Herbsttagung**
und 49. Mitgliederversammlung von DANTE e.V.
Köln
9. 11. – 10. 11. **OpenRheinRuhr**
Freie Software und Netzpolitik
Rheinisches Industriemuseum (RIM) Oberhausen
<http://www.openrheinruhr.de/>

2014

- April **DANTE 2014**
und 50. Mitgliederversammlung von DANTE e.V.
Universität Heidelberg
25. 6. – 28. 6. **Linuxtag Berlin**
Messegelände
14055 Berlin
<http://www.linuxtag.org/>

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

Michael-E. Voges, Tel.: (03362) 50 18 35,
mevoges@t-online.de
Mantee – Café Restaurant, Chausseestraße 131, 10115 Berlin
Zweiter Donnerstag im Monat, 19.00 Uhr

Bielefeld

Jürgen Schwarze, Tel.: (0521) 5 57 39 06,
juergen.schwarze@bitel.net
Ferdis Pizza Pinte, Schmiedestraße 9, 33613 Bielefeld,
Zweiter Montag im Monat, 19.30 Uhr

Bremen

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

Frankfurt

Harald Vajkonny,
<http://wiki.lug-frankfurt.de/TeXStammtisch>
Restaurant »Zum Jordan«, Westerbachstr. 7, 60489 Frankfurt
Zweimonatlich, Vierter Donnerstag im Monat, 19.30 Uhr

Hamburg

Lothar Fröhling,
lothar@thefroehlings.de
Restaurant Sandstuv, 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

Uwe Ziegenhagen

Ersatzbau Geowissenschaften, Greinstr. 6, 50674 Köln

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,

ttt@schoenerpublizieren.de

Anmeldeseite zur Mailingliste: <http://lists.schoenerpublizieren.de/cgi-bin/mailman/listinfo/ttt>

Universität Trier

Monatlich

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 \TeX 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

IBAN DE67 6709 0000 0002 3100 07

Kontonummer 2 310 007

SWIFT-BIC GENODE61MA2

Präsidium

Präsident: Martin Sievers president@dante.de

Vizepräsident: Herbert Voß vice-president@dante.de

Schatzmeister: Klaus Höppner treasurer@dante.de

Schriftführer: Manfred Lotz secretary@dante.de

Beisitzer:
Patrick Gundlach
Volker RW Schaa
Uwe Ziegenhagen

Ehrenmitglieder

Peter Sandner 22.03.1990 Klaus Thull † 22.03.1990

Yannis Haralambous 05.09.1991 Barbara Beeton 27.02.1997

Luzia Dietsche 27.02.1997 Donald E. Knuth 27.02.1997

Eberhard Mattes 27.02.1997 Hermann Zapf 19.02.1999

Server

DANTE: <http://www.dante.de/> (Rainer Schöpf, Joachim Schrodt)

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

FAQ

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

\TeX : <http://projekte.dante.de/DanteFAQ/WebHome>

Autoren/Organisatoren

Karl Berry T _E XLive http://tug.org/texcollection	[13]	Dick Koch MacT _E X http://tug.org/texcollection	[13]
Marco Daniel Beerentaltrift 90g 21077 Hamburg marco.daniel@mada-nada.de	[28]	Markus Kohm Freiherr-von-Drais-Straße 66 68535 Edingen-Neckarhausen komascript@gmx.info	[16]
Jürgen Fenn Friedensallee 174/20 63263 Neu-Isenburg juergen.fenn@gmx.de	[43]	Manfred Lotz CTAN http://tug.org/texcollection	[13]
Thomas Feuerstack proT _E Xt http://tug.org/texcollection	[13]	Herb Schulz MacT _E Xtras http://tug.org/texcollection	[13]
Falk Hohlfeld fachho@aol.com	[9]	Martin Sievers siehe Seite 50	[4, 7]
Roger Jud Schorenstrasse 1 9000 St. Gallen, Schweiz rogerjud@gmx.ch	[24, 30]	Herbert Voß Wasgenstraße 21 14129 Berlin herbert@dante.de	[3, 37, 46]
		Uwe Ziegenhagen uwe@ziegenhagen.info	[7]

Die T_EXnische Komödie

25. Jahrgang Heft 3/2013 August 2013

Impressum

Editorial

Hinter der Bühne

- 4 Grußwort
- 7 Einladung zur Herbsttagung 2013 in Köln

T_EX-Theatertage

- 9 Messegeflüster
- 13 Die DVD mit der T_EX-Collection 2013

Bretter, die die Welt bedeuten

- 16 Was ist eigentlich: die Besonderheit des @-Zeichens in Befehlsnamen?
- 24 Autovervollständigung mit T_EXnicCenter

Tipps und Tricks

- 28 Das Paket `minted` und der Apostroph
- 30 Eulersche Gerade mit `tkz-euclide` zeichnen
- 37 Schleifenmakro
- 38 Mathematikschriften – Vergleich

Von fremden Bühnen

- 43 Neue Pakete auf CTAN
- 46 Im Netz gefunden

Spielplan

- 47 Termine
- 48 Stammtische

Adressen

- 51 Autoren/Organisatoren