

Die T_EXnische Komödie

dante
Deutschsprachige
Anwendervereinigung T_EX e.V.

32. Jahrgang Heft 3/2020 August 2020

3/2020

Impressum

»Die T_EXnische Komödie« ist die Mitgliedszeitschrift von DANTE e.V. Der Bezugspreis ist im Mitgliedsbeitrag enthalten. Namentlich gekennzeichnete Beiträge geben die Meinung der 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-L^AT_EX-Quellcode unter Verwendung der Dokumentenklasse dtk erstellt und per E-Mail oder Datenträger (z. B. CD/DVD) an unten stehende Adresse der Redaktion geschickt werden. Sind spezielle Makros, L^AT_EX-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 <https://projekte.dante.de/DTK/AutorInfo> von DANTE e.V.

Diese Ausgabe wurde mit LuaHBTeX, Version 1.13.0 (TeX Live 2021/dev) erstellt. Als Standardschriften kamen Libertinus Serif, Libertinus Sans Serif, Anonymous Pro und Libertinus Math zum Einsatz.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 2400

Herausgeber: DANTE, Deutschsprachige Anwendervereinigung T_EX e.V.
Postfach 11 03 61
69072 Heidelberg

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

Druck: Schleunungdruck GmbH
Eltertstraße 27, 97828 Marktheidenfeld

Redaktion: Herbert Voß (verantwortlicher Redakteur)

Mitarbeit: Lukas C. Bossert Eberhard Lisse Rolf Niepraschk
Günter Partosch Christine Römer Martin Sievers

Editorial

Liebe Leserinnen und liebe Leser,

obwohl wir aus allseits bekannten Gründen keine Mitgliederversammlung abhalten konnten, gibt es einige offizielle Dinge: Den Bericht der Kassenprüfer und der Kassenwartin. Genau genommen müsste ich schreiben: Den Bericht der Kassenprüferin und der Kassenprüfer, um es halbwegs gendergerecht zu gestalten. »Die $\TeX$ nische Komödie« hat es mit diesen formalen Dingen bislang nicht so genau genommen bzw. keine Vereinheitlichung vorgenommen, wie auch im vorliegenden Heft festgestellt werden kann. Ansonsten orientiert sich die Redaktion an der amtlichen Rechtschreibung und um eine orthografische Homogenität. Der Rat für deutsche Rechtschreibung hat unserer Kenntnis nach noch keine Regelung für eine geschlechtergerechte Schreibung vorgenommen. Mit Binnen-I, Genderstern, Gendergap, Beidbenennungen, Klammern, Schrägstrichen etc. sind verschiedene Varianten im Gebrauch.

Während ich überhaupt keine Probleme damit habe, meine Studentinnen und Studenten als »Studierende« anzusprechen, ist dies bei »Frau Prof.in Dr.in Lise Meitner« doch anders.¹ Auch ist meine gewohnte Anrede in diesem Editorial mittlerweile nur noch halbwegs korrekt, denn sie geht von der binären Geschlechterordnung aus. Um dieses Thema abzukürzen: Gendergerechte Sprache will geübt und vor allem auch »gewollt« werden. Beim letzten Punkt besteht bei mir vielleicht noch Änderungsbedarf ...



Sie finden in dieser Ausgabe weiterhin einen Beitrag von Alexander Krumeich, der eine interessante Anwendung von $\text{\LaTeX}$ beschreibt. Pascal Braband beschreibt seine von ihm entwickelte Software, die die Ausgabe eines informellen Bereiches bei einer Beamerpräsentation ermöglicht. Auf einem zweiten Bildschirm kann man sich zusätzliche Informationen anzeigen lassen, die nicht Teil der eigentlichen Präsentation sind.

Ich wünsche Ihnen wie immer viel Spaß beim Lesen und verbleibe

mit $\TeX$ nischen Grüßen

Ihr Herbert Voß

¹ Siehe <https://frauenbeauftragte.hu-berlin.de/de/informationen/geschlechtergerechte-sprache/leitfaden-geschlechtergerechte-sprache-humboldt.pdf>, S. 18.

Hinter der Bühne

Vereinsinternes

Grußwort

Liebe Mitglieder,

»Binge-Watching« bezeichnet »stundenlanges Anschauen mehrerer Folgen einer Serie hintereinander« (Quelle: <https://www.duden.de/node/239317/revision/242794>). Schon vor Beginn der coronabedingten Einschränkungen hat so mancher das eine oder andere Wochenende mit »bingen« verbracht. Durch den Wegfall zahlreicher anderer Freizeitaktivitäten hatten Streamingdienste wie Netflix oder Amazon Prime großen Zulauf und drosselten nach Gesprächen mit der EU-Kommission zeitweilig sogar ihre Übertragungsraten, um die Netze nicht zu überlasten.

Wer wie ich Kinder hat, hatte in den vergangenen Monaten sicherlich andere Sorgen als die Frage nach der nächsten Serie zum »Durchschauen«. Der Zufall wollte es nun aber, dass ich am vergangenen Wochenende Gelegenheit hatte, die virtuelle TUG-Konferenz zu »besuchen«. Ich möchte vorweg schicken, dass ich großen Respekt vor den Aufgaben des Organisationsteams unter Leitung von Paulo Ney de Souza habe und es toll finde, was für ein reichhaltiges, weltweites und hochkarätiges Vortragsprogramm sie zusammengestellt haben.

Nach einem Einführungsworkshop am Donnerstag startete mein persönliches Binge-Watching-Experiment am Freitag um 18 Uhr MESZ mit einer sehr interessanten Keynote über die Entwicklung der Noto-Fonts, mit denen Google alle Zeichen des Unicodestandards abdecken möchte. Bis 2:30 Uhr am nächsten Morgen folgten bunt gemischte Beiträge nicht nur zum Thema »Schriften«.

Am Samstag begann mein Tagungstag um 9 Uhr, passenderweise mit einem »Coffeetalk«. Den hatten die amerikanischen Teilnehmer vermutlich auch nötig, war es bei ihnen doch gerade Nacht geworden. Am anderen Ende der Welt, in Australien, konnten die Teilnehmer hingegen bereits einen Feierabendkaffee zu sich nehmen. Mit der Universität von Adelaide hatte die TUG einen Sponsor für eine Zoom-Webinarlizenz gewinnen können. Dadurch waren nur die Referentinnen und Referenten per Video zugeschaltet, alle anderen nur als Zuhörer.

Die Firma Zoom gehört mit ihrem gleichnamigen Produkt für Videokonferenzen sicherlich zu den Gewinnern der Coronazeit und aus technischer Sicht macht sie

auch sehr vieles richtig. Kritik gab und gibt es aber im Bezug auf den Datenschutz. Die TUG sah jedoch nach eigener Aussage keine Alternative zu Zoom als primärem Veranstaltungsort. Gleichzeitig wurden alle Vorträge auch auf YouTube live ausgestrahlt und werden zudem im Nachgang als Videos zum Abruf bereitgestellt. Für den informellen Austausch zwischendurch, der bei solchen Tagungen einen nicht zu unterschätzenden Wert hat, standen Chaträume über Zulip sowie eine »Gather Town« zur Verfügung. Beides für mich Neuland, das ich mir wohl in Zukunft näher ansehen werde.

Das Angebot für all diese Kanäle einer kostenfreien virtuellen T_EX-Konferenz, von denen es in diesem Jahr ja nur ganz wenige gibt, wurde von mehr als 200 Personen angenommen. Zusammen mit den YouTube-Zuschauern und den »Klicks« im Nachgang kann man mit Fug und Recht von einem gelungenen Experiment sprechen. Es gab zwar vereinzelt die üblichen Phänomene bei Videogesprächen von »Hört man mich?«, »Könnt ihr mich sehen?«, »Dein Mikro ist aus.«, »Dein Mikro ist noch an.« über ungeplantes Teilen von Bildschirmhalten bis zum »Zoombombing«, also der Störung eines Vortrags durch eine fremde Person. Auch das Abspielen der vorproduzierten Videos per »Bildschirm teilen«-Funktion funktionierte nicht immer zufriedenstellend und machte das aufmerksame Verfolgen der Vorträge manchmal etwas schwierig. Doch wie eingangs gesagt: Ich habe großen Respekt davor, was hier auf die Beine gestellt wurde und bin froh, dass auch einige von Ihnen die Möglichkeit genutzt haben, als Zuschauer, aber auch als Beitragende mit dabei zu sein. Auch für künftige »normale« TUG-Tagungen stehen die Chancen gut, dass es jeweils auch eine digitale Komponente geben wird.

Zurück zum Programm am Samstag: Es ging an diesem Tag viel um die angemessene Vermittlung von T_EXnischem Wissen sowie neue Plattformen und Angebote, um dies zu erreichen. Auch die nahe und fernere Zukunft von L^AT_EX3 waren Thema. Der Tag endete (für mich) gegen 22 Uhr mit einem Interview mit Philip Kime, das wie schon sein Vortrag auf der letztjährigen Herbsttagung sehr unterhaltsam war. Die Vorträge zu nachtschlafender Zeit ließ ich dann aus, so dass das Programm am Sonntag für mich erst am Nachmittag begann. Bis zum späten Abend spielten moderne webbasierte Anwendungen von T_EX bzw. Ergänzungen für mobile Geräte sowie die Verknüpfung von T_EX mit der modernen IT-Welt die Hauptrolle. Mein Binge-Watching-Experiment endete Montag früh gegen 0:30 Uhr mit vielen Eindrücken und neuen Erkenntnissen. Ich hoffe, ich konnte Ihnen einen guten Einblick geben und Sie neugierig machen auf die Videos und die zugehörigen Konferenzartikel, die in traditioneller Form in einer der nächsten TUGboat-Ausgaben erscheinen werden.

Zunächst aber wünsche ich Ihnen viel Vergnügen bei der weiteren Lektüre.

Herzlichst Ihr/Euer
Martin Sievers

Bericht der Rechnungsprüfer zum Vereinsjahr 2019

Mario Haustein, Stephan Lukasczyk, Marei Peischl

Einführung

Am Samstag, den 29. Februar 2020, trafen sich die von den Vereinsmitgliedern ordentlich gewählten Rechnungsprüfer Mario Haustein, Stephan Lukasczyk und Marei Peischl zur Rechnungsprüfung im Vereinsbüro in Heidelberg. Neben der Büroleiterin Karin Dornacher war auch die Schatzmeisterin Doris Behrendt anwesend. Die Prüfung des Rechnungsjahres 2019 begann um 10:57 Uhr und endete am selben Tag um ca. 15:15 Uhr. Nach ersten allgemeinen Informationen über das vergangene Rechnungsjahr wurden alle zur Prüfung notwendigen Akten nebst Kontenplänen von Karin Dornacher zur Überprüfung vorgelegt.

Rechtsgrundlage

Die Rechnungsprüfung wurde nach den GOB¹ und den Zielen des Vereins nach der derzeit gültigen Vereinssatzung durchgeführt.

Die Ergebnisse der Prüfung wurden unmittelbar protokolliert.

Prüfungsgegenstände

- Barkasse
- Bankkonten bei VR-Bank und Degussa-Bank
- PayPal-Abrechnungskonto
- Bankauszüge
- Kreditoren (Rechnungen)
- Spesen-Abrechnungen
- Inventar und Buchbestand

Eröffnungs- und Endstände der Bank- und Buchhaltungskonten

Die Stände der Eröffnungsbuchungen zum 31. Dezember 2018 bzw. 1. Januar 2019 (nach Abschluss des Vereinsrechnungsjahres 2018) sowie für den Jahresabschluss des Rechnungsjahres 2019 sind in nachfolgender Tabelle aufgeführt:

¹ Kaufm. Grundsätze der ordnungsgemäßen Buchführung

Buchungs- konto intern	Bezeichnung	Endziffern Kontonr. Bank	Schlussbestand 31.12.2018	Schlussbestand 31.12.2019
001	Barkasse	–	559,64 €	438,09 €
011	VR-Bank Giro	007	11.682,44 €	4915,43 €
012	VR-Bank Cash	011	50.236,45 €	43.235,78 €
020	VR-Extrazins IV	861	10.167,63 €	10.168,48 €
031	PayPal	–	333,14 €	810,60 €
041	Degussa-Bank	726	35.552,85 €	35.556,45 €

Die Eröffnungsbuchung des Kontos *VR-Bank Giro* wich in der Buchhaltung (Buchungskonto 011) aufgrund einer unzutreffenden Stornobuchung um 23,49 € ab. Dies ist eine Korrektur der Buchhaltung von 2018, vergleiche Kassenprüfungsbericht in der DTK 2/2019 [1].

Feststellungen

Die Salden der Buchführung stimmen mit den Bankauszügen überein, die Anfangsstände 2019 entsprechen, bis auf obige Abweichung, den Endständen 2018. Die jeweiligen SBK²- und EBK³-Buchungen wurden ordnungsgemäß durchgeführt.

Die Belege wurden in folgendem Umfang geprüft: Bei den Konten 451 (lfd. Kosten Büro), 452 (Lohnkosten), 455 (Portokosten), 460 (Webserver), 820 (Spendeneingänge), 822 (sonstige Verkaufserträge), 850 (Zinserträge) und 890 (sonstige Einnahmen) wurde durch Stichproben geprüft. Die Eingangskonten 810 und 815 (Beiträge DANTE bzw. TUG) haben wir überschlägig geprüft (Anzahl der Vereinsmitglieder, eingegangene Beiträge). Alle Belege und Buchungen wurden geprüft bezüglich der Konten 430 (Reisekosten & Spesen, Vorstand) und 480 (Reisekosten & Spesen, andere), 410 (Komödie, Druck, Versand), 441 (Tagung Frühjahr), 442 (Tagung Herbst), 446 (Messen) sowie 447 (Hackercamp).

Der Bücherbestand ist zum Zeitpunkt der Kassenprüfung nicht prüfbar, da teilweise Bücher für kommende Messen verpackt wurden.

Alle Konten der Buchführung sind in den Jahresabschluss eingeflossen. Laut mündlicher Erklärung unserer Schatzmeisterin Doris Behrendt existieren keine weiteren Konten auf den Namen des Vereins.

² Schluss-Bilanz-Konto = Abschlussbuchung eines Geschäftsjahres

³ Eröffnungs-Bilanz-Konto = Erstbuchung eines Geschäftsjahres (Übertragsbuchung)

Barkasse

Die Barbelege und der Kassenbestand wurden ebenfalls geprüft. Ausgehend vom Schlussbestand der Barkasse zum 31.12.2019 berechneten wir unter Berücksichtigung aller Barbuchungen seit diesem Zeitpunkt den Sollstand der Kasse zum Zeitpunkt der Kassenprüfung. Der Sollstand stimmte mit dem vorgefunden Iststand überein.

Leider besaßen einige Barbelege doppelte bzw. falsch vergebene Belegnummern. Zudem stellten wir entsprechend abweichende Belegnummern zwischen Beleg und Kontoauszug fest. Alle Unklarheiten ließen sich im Rahmen der Kassenprüfung aufklären und wurden umgehend korrigiert. Sie sind daher kein Grund zur Beanstandung. Wir empfehlen, falsch gebuchte Belege deutlich als fehlerhaft zu kennzeichnen, um die Korrekturbuchungen besser nachvollziehen zu können.

Anmerkungen

Insgesamt konnten sämtliche geprüften Vorgänge nachvollzogen werden. Allerdings war dies bei einigen erst nach detaillierter Nachfrage möglich. Wir haben die erfragten Informationen entsprechend vermerkt, sodass dies nun auch ohne Rückfrage möglich ist. Allerdings möchten wir für die Zukunft darum bitten, auf die Nachvollziehbarkeit der gesamten Vorgänge zu achten.

Positiv möchten wir die Belege zu den Projektförderungen hervorheben. Dort sind zu jeder Ausgabe sämtliche Einzelbelege vorhanden:

- Vorstandsbeschluss; meistens in Form der E-Mail, mit der dem Antragssteller die Genehmigung mitgeteilt wurde;
- Belege über die tatsächlichen Ausgaben, die entweder ganz oder anteilig erstattet werden;
- Ggf. eine kommentierte Nebenrechnung, sodass auch die anteilige Zahlung nachvollzogen werden kann;
- Belege über geflossene Zahlungen.

So ist es ohne zusätzliche Rückfragen möglich, die Ausgabe als satzungskonform und im Sinne der Wirtschaftlichkeit zu bescheinigen.

Leider müssen wir auch einige Punkte bemängeln. Bei einigen Reisekostenabrechnungen fehlten leider die Anlagen zum Verpflegungsmehraufwand. Diese sind zwar mit den vorhandenen Daten nachzuvollziehen, dennoch bitten wir darum, dass jeder, der diese Kosten geltend macht, entsprechende Unterlagen möglichst vollständig einreicht. Dies erleichtert die Buchhaltung und insbesondere die spätere Prüfung ungemein.

Selbiges gilt, falls die tatsächlichen Ausgaben des Abrechnenden höher waren, als der abgerechnete Kostenanteil. Dies ist beispielsweise der Fall, wenn das gekaufte Zugticket eine Fahrkarte erster Klasse ist. Der Verein erstattet aus Gründen der Wirtschaftlichkeit nur Tickets zweiter Klasse. Um den Vorgang nachvollziehbar zu halten, empfehlen wir, sowohl die Fahrkarte erster Klasse zum Nachweis der Fahrt, als auch einen Eigenbeleg über die abgerechneten Kosten beizufügen. Der Eigenbeleg sollte den abgerechneten Fahrpreis, das heißt, den Vergleichspreis zweiter Klasse ausweisen. Zusätzlich zum Eigenbeleg sollte ein Screenshot mit nachvollziehbarem Erstellungszeitpunkt den Vergleichspreis zweiter Klasse belegen.

Wir haben zwischen den Reisen und der Abrechnung der Reisekosten teilweise erhebliche Zeiträume festgestellt. Wir bitten darum, Reisekosten möglichst zeitnah abzurechnen, da es nur so möglich ist, die Kosten für Veranstaltungen in einem Gesamtbild zu betrachten. Insbesondere sollte es nach Möglichkeit vermieden werden, Reisekosten aus dem Vorjahr erst im Folgejahr abzurechnen!

So war das Chaos Communication Camp 2019 zum Zeitpunkt der Kassenprüfung noch nicht vollständig abgerechnet. Das Konto 447 (Hackercamp) umfasst zum Zeitpunkt der Prüfung also noch nicht den gesamten Kostenumfang der Veranstaltung. Dies machte es unmöglich, einen Überblick über die tatsächlichen Kosten zu erhalten. Wir haben uns diesen Punkt für das nächste Jahr vorgemerkt.

Außerdem möchten wir anmerken, dass einige Bewirtsungsbelege nicht den rechtlichen Vorgaben entsprachen. So fehlten teilweise Unterschriften, explizite Ausweisung von Trinkgeldern, oder Teilnehmerlisten. Bei dieser Gelegenheit möchten wir außerdem darauf hinweisen, dass mit dem gängigen Thermodruckpapier die Lesbarkeit über die gesamte Aufbewahrungsdauer der Belege kaum gegeben ist. Hier empfehlen wir eine Fotokopie des Beleges beizuheften; einige Belege waren bereits nach weniger als einem Jahr kaum mehr entzifferbar.

Ferner bitten wir darum, bei Belegen in ausländischen Währungen (insbesondere TUG-Reisekosten) die Wechselkurse oder den tatsächlich bezahlten Eurobetrag aufzuführen. Dies kann zum Beispiel durch einen entsprechenden Eigenbeleg, zusammen mit einem Auszug der entsprechenden Kreditkartenabrechnung, passieren, um die Belege nachvollziehbar zu halten.

Bei der Prüfung der Spendenquittungen konnte keine Beanstandung, außer einem fehlenden Wortzwischenraum, festgestellt werden. Auch diese Korrektur wurde uns zugesichert.

Freistellung

Am 18.10.2018 hat das Finanzamt Heidelberg einen neuen Freistellungsbescheid zur Körperschaftssteuer und Gewerbesteuer für die Jahre 2015 bis 2017 ausgestellt. Die

damit verbundene Anerkennung als steuerbegünstigter und gemeinnütziger Verein berechtigt den Verein u. a. auch dazu, sog. Zuwendungsbestätigungen auszustellen. Der nächste Antrag ist, nach heutiger Rechtslage, im Jahr 2021 für die Jahre 2018 bis 2020 zu stellen. Er gilt – sofern bewilligt – wieder rückwirkend für drei Jahre.

Die Ausgaben des Vereins erfolgten satzungskonform und stets nach dem Grundsatz der Wirtschaftlichkeit. Die Führung eines Vereins mit knapp 2000 Mitgliedern und internationalen Verbindungen verursacht naturgemäß einen gewissen Aufwand.

Abschluss

Die Anfangs- und Endbestände der Konten stimmen mit den Kontoauszügen, welche lückenlos und vollständig vorhanden sind, überein. Die Rechnungen und Journale der Unterkonten sind vollständig. Sämtliche vorhandenen Unterlagen des Rechnungsjahres sind ordentlich abgeheftet.

Alle Geschäftsvorfälle wurden in der Buchhaltung richtig und konsistent kontiert und können nun mithilfe der Anmerkungen klar in den Kontoauszügen und Rechnungen nachverfolgt werden.

Der Schatzmeisterin Doris Behrendt und der Büroleiterin Karin Dornacher wird eine ordentliche Buch- und Kassenführung bescheinigt. Bei allen Geschäftsvorfällen blieben die Vereinsinteressen gewahrt.

Heidelberg, den 29. Februar 2020

Mario Haustein, Stephan Lukasczyk, Marei Peischl

Literatur und Software

- [1] Stephan Lukasczyk, Alexander Willand: »Bericht der Rechnungsprüfer zum Vereinsjahr 2018«, *Die T_EXnische Komödie*, 31.2 (2019), 13–15.

Vorläufiger Bericht der Schatzmeisterin für das Jahr 2019

Doris Behrendt

Ich möchte Sie, auch wenn es dieses Jahr noch keine Mitgliederversammlung gegeben hat, über den aktuellen Stand der Finanzen informieren.

Alle in den folgenden Tabellen erscheinenden Daten sind entweder Stückzahlen, also natürliche Zahlen, oder Eurobeträge. Die Daten, die von vor der Eurozeit stammen, wurden von DM zu € mit 2 : 1 umgerechnet.

Bankkonten/Finanzkonten

Neben der Barkasse hat DANTE e.V. drei Konten bei der Volksbank (VR Bank Rhein-Neckar eG), ein PayPal-Konto und ein Firmentagesgeldkonto bei der Degussa Bank. Die angegebenen Kontonummern (Kst. für *Kostenstelle*) beziehen sich auf die interne Nummerierung in der Buchhaltung.

Tab. 1: Bankkonten

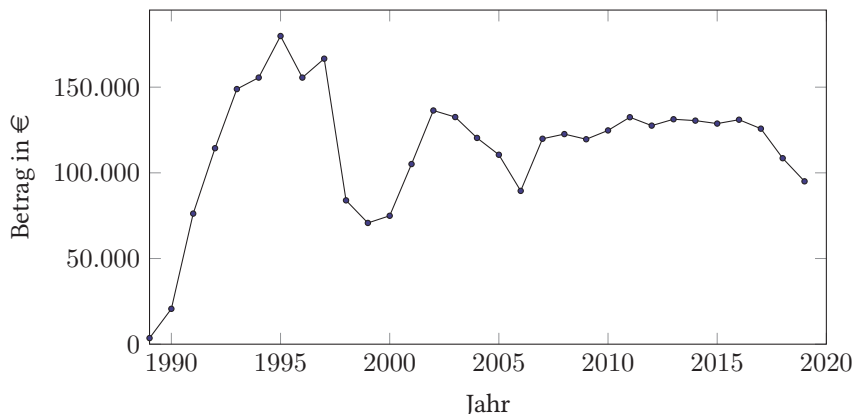
Kst.	Konto	Stand 01.01.2019	Stand 31.12.2019	Saldo
001	Kasse	559, 64	438, 09	−121, 55
011	VR-Vereinsgirokonto	11.682, 44	4.915, 43	−6.767, 01
012	VR-Cash	50.236, 45	43.235, 78	−7.000, 67
020	VR-Mehrzinssparen	10.167, 63	10.168, 48	0, 85
031	Paypal	333, 14	810, 60	477, 46
041	Degussa	35.552, 85	35.556, 45	3, 60
Summe		108.532, 15	95.124, 83	−13.407, 32

Wie man sieht, haben sich unsere Rücklagen um gut 13.000 € verringert.

Eine Darstellung der Summe der Kontostände aller Bankkonten zum jeweiligen Jahresende seit 1989 liefert die Abbildung 1 auf der nächsten Seite¹.

¹ Rekonstruiert aus den Finanzberichten der vergangenen Jahre, jeweils in der T_EXnischen Komödie, vor 2001 auf € umgerechnet

Abb. 1: Summe der Kontostände am Jahresende



Kostenstellenübersicht

Die einzelnen Einnahmen und Ausgaben werden auf unterschiedliche Kostenstellen verbucht, die im Folgenden dargestellt sind. Diese sind in Kostenstellen für Einnahmen (800 und folgende) sowie für Ausgaben (400 und folgende) aufgeteilt. Zum Vergleich sind die Werte aus verschiedenen Vorjahren mit angegeben.

Einnahmen im Vereinsjahr 2019

Die Einnahmen, die in Tabelle 2 auf der nächsten Seite dargestellt sind, kommen zum größten Teil aus den Mitgliedsbeiträgen.

Da einige Mitglieder ihren Mitgliedsbeitrag für das Mitgliedsjahr n oft schon im Dezember des Jahres $n - 1$ überweisen oder vereinzelt auch erst im Jahr $n + 1$, muss man den Betrag der Kostenstelle 810 zur Summe der Beträge der Kostenstelle 812 und 813 für das Folgejahr bzw. Vorjahr hinzuaddieren, um auf die für das jeweilige Kalenderjahr eingenommenen Mitgliedsbeiträge zu kommen (siehe Tabelle 3 auf der nächsten Seite). Dieses Mal fällt der Betrag, der schon 2019 für 2020 überwiesen wurde, um ca. 7000 € geringer aus als im Vorjahr, weil ich die Rechnungen für die Mitgliedsbeiträge 2020 erst heuer im Frühjahr verschickt habe. Vorher waren sie immer Ende Dezember verschickt worden.

Über die letzten zehn Jahre gerechnet sind unsere Einnahmen durch Mitgliedsbeiträge um ca. 10 % gesunken. Dies zeigt auch Abbildung 2 auf Seite 14.

Tab. 2: Einnahmen

Kst.	Bezeichnung	2019	2018	2014	2012
810	Mitgliedsbeiträge aktuelles Jahr	68.859, 61	69.872, 00	77.225, 00	78.829, 46
812	Mgl.beiträge für Vorjahr	40, 00	80, 00	0	60, 00
813	Mgl.beiträge für Folgejahre	1.138, 00	7.445, 00	9.935, 00	5.846, 50
815	Mgl.beiträge TUG (Doppelmgl.schaft)	1.947, 00	1.958, 50	1.744, 50	2.066, 00
820	Spenden	3.355, 95	3.196, 55	2.556, 29	2.026, 70
822	Verkauf sonstiges	744, 00	946, 35	4.333, 85	298, 50
830	Verkauf Bücher	2.198, 85	5.397, 74	2.864, 10	6.893, 70
841	DANTE Frühjahrstagung	1.125, 00	1.370, 00	2.610, 00	540, 00
842	DANTE Herbsttagung	0	0	0	0
850	Zinsen	3, 78	5, 64	860, 67	1.763, 61
860	WinEdt	60	46, 50	111, 50	287, 00
890	Einnahmen sonstiges	-49, 56	-99, 49	53, 44	133, 61
Summe		79.422, 63	90.218, 79	102.294, 35	98.745, 08

Tab. 3: Mitgliedsbeiträge für das Jahr n

Jahr n	im Jahr n gezahlt	im Vorjahr gezahlt	im Folgejahr gezahlt	Summe
2008	62.647, 13	21.879, 73	140, 00	84.666, 86
2009	72.793, 87	13.126, 27	40, 00	85.960, 14
2010	82.354, 19	1.670, 00	80, 00	84.104, 19
2011	73.942, 55	9.478, 50	60, 00	83.481, 05
2012	78.829, 46	6.265, 50	160, 00	85.254, 96
2013	77.605, 00	5.846, 60	0, 00	83.451, 60
2014	77.225, 00	5.855, 00	0, 00	83.080, 00
2015	71.956, 00	9.895, 00	0, 00	81.851, 00
2016	74.306, 96	6.430, 10	80, 00	80.817, 06
2017	77.618, 90	1.680, 00	80, 00	79.378, 90
2018	69.872, 00	8.528, 00	40, 00	78.440, 00
2019	68.859, 61	7.445, 00		76.304, 61

Ausgaben

Demgegenüber stehen die Ausgaben, dargestellt in Tabelle 4 auf der nächsten Seite. Ihre Nummer im Kontenrahmen der Buchhaltungssoftware wird – bis auf den Projektfonds – von einer 4 angeführt.

Die größten Ausgabenposten ins Verhältnis gesetzt zeigt auch noch Abbildung 3 auf Seite 15.

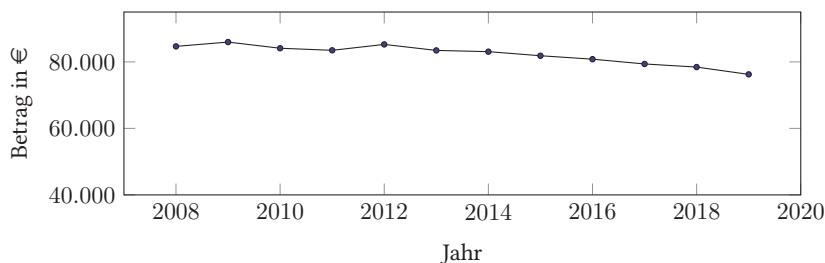


Abb. 2: Mitgliedsbeiträge in den vergangenen Vereinsjahren

Tab. 4: Ausgaben

Kst.	Bezeichnung	2019	2018	2014	2012	2002
410	Komödie (Druck, Versand)	19.584,86	18.544,12	18.704,91	18.246,55	14.617,37
415	TUG-Beiträge	2.009,26	2.001,67	1.601,30	2.057,10	0
420	Einkauf Bücher	720,53	2.706,17	1.416,38	3.068,14	48,39
422	Einkauf sonstiges	0,00	23,80	423,48	165,00	0
423	T _E XLive DVD	1.707,77	1.689,92	11.943,06	1.883,25	0
430	Vorstand (Spesen)	13.690,02	16.353,77	10.896,85	13.647,81	4.150,04
431	Ehrungen	500,00	500,00	0	252,00	0
441	Frühjahrstagung	1.820,00	1.994,38	8.279,16	814,28	1.419,04
442	Herbsttagung	830,57	400,29	297,30	0	1.330,42
443	Bursary f. andere Tag.	940,53	1.816,00	1.000,00	1.000,00	5.161,87
446	Messen (CLT etc.)	2.890,37	2.564,67	2.154,86	2.363,85	0
447	Hackercamp/Zeltlager	3.802,39	0	0	0	0
448	neu: Versicherungen	481,96	481,96			
451	Büro Miete, Nebenkosten	9.388,97	10.032,62	9.130,35	8.766,11	11.304,06
452	Büro Personal	24.404,49	32.161,51	29.302,84	29.004,74	24.701,18
453	Verbrauchsmaterial	377,32	467,16	233,05	288,17	584,81
454	Inventar/Anschaffungen	890,17	285,74	497,78	651,27	0
455	Porto	1.003,99	2.468,17	2.217,24	2.317,86	1.209,89
460	Web-Server	1.717,26	1.828,96	734,61	2.410,17	643,94
473	Spenden	0	0	100,00	0	0
480	Spesen (nicht Vorstand)	2.509,02	2.027,10	973,62	1.190,45	1.563,45
481	Sonstiges	57,00	0	0	0	0
485	Geschenke	0	0	242,00	145,50	302,11
499	Gebühren	203,68	228,87	129,58	153,57	230,28
6474	Projektfonds	3.299,79	8.890,03	2.931,47	13.800,34	0
	Summe	92.829,95	107.466,91	103.209,84	103.226,16	68.913,29

Vergleicht man die einzelnen Ausgabenposten mit den Werten aus den Vorjahren, so weisen die meisten Kostenstellen keine großen Schwankungen auf.

Im Jahr 2019 gab es wieder ein Hackercamp, im Jahr 2018 nicht. Das erste Hackercamp in den Niederlanden im Jahr 2017 lag bei Ausgaben von ca. 4600 €.

Aus dem Projektfonds sind im Jahr 2018 knapp 9.000 € an verschiedene Projekte ausgezahlt worden (siehe hierzu Tabelle 7 auf Seite 18), im Jahr 2019 wurden weniger Projektmittel abgerufen, als eingeplant.

Beim Büropersonal konnten wir Kosten einsparen, indem wir die Präsenzzeiten um einige Stunden verringert haben.

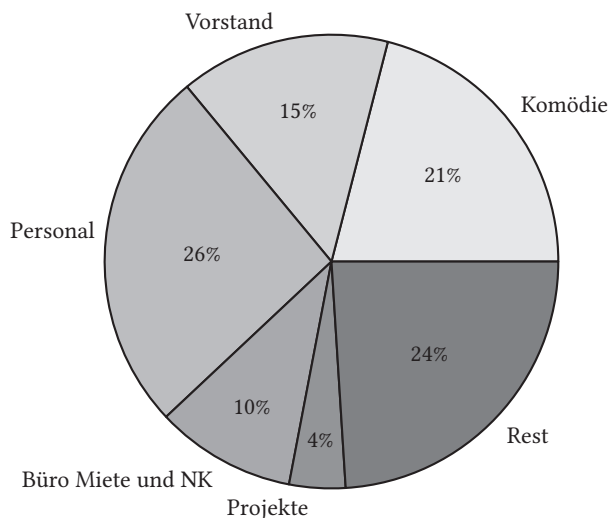


Abb. 3: Zusammensetzung der Ausgaben im Vereinsjahr 2019

Saldo der Einnahmen und Ausgaben

Eine Gegenüberstellung der Gesamteinnahmen und -ausgaben findet sich in Tabelle 5 auf der nächsten Seite. Ich habe dazu alle alten Ausgaben der T_EXnischen Komödie durchsucht und für die Jahre vor 2001 die DM-Beträge in € umgerechnet. Genauer dazu finden Sie im Finanzbericht für das Jahr 2018 in der DTK 19-2.

Abgleich mit Plan für 2019

Die Planung für das Jahr 2019 lag auf der Einnahmenseite knapp 7.000 € unter den tatsächlichen Einnahmen (siehe Tabelle 6 auf Seite 17). Der Hauptgrund dafür ist, dass die Rechnungen für die Mitgliedsbeiträge 2020 erst heuer im Frühjahr verschickt wurden und nicht schon kurz vor Weihnachten 2019. Deshalb wurden dieses Mal fast keine Beitragszahlungen für 2020 bereits 2019 verbucht.

Tab. 5: Gegenüberstellung der Einnahmen und Ausgaben

<i>Jahr</i>	<i>Einnahmen</i>	<i>Ausgaben</i>	<i>Saldo</i>
2019	79.422, 63	92.829, 95	−13.407, 32
2018	90.218, 79	107.466, 91	−17.248, 12
2017	97.797, 62	103.016, 48	−5.218, 86
2016	87.486, 13	85.217, 98	2.268, 15
2015	96.977, 18	98.702, 07	−1.724, 89
2014	102.147, 47	102.949, 84	−802, 37
2013	96.663, 70	92.938, 86	3.724, 84
2012	98.322, 93	103.226, 16	−4.903, 23
2011	91.099, 10	83.399, 49	7.699, 61
2010	103.814, 10	98.789, 75	5.024, 35
2009	87.972, 16	90.983, 98	−3.011, 82
2008	88.611, 94	85.889, 95	2.721, 99
2007	113.853, 20	83.436, 61	30.416, 59
2006	76.749, 39	97.826, 58	−21.077, 19
2005	107.218, 74	117.387, 25	−10.168, 51
2004	102.787, 87	114.961, 42	−12.173, 55
2003	89.167, 95	93.002, 79	−3.834, 84
2002	99.027, 45	67.761, 92	31.265, 53
2001	104.901, 44	74.677, 27	30.224, 17
2000	95.301, 56	91.629, 57	3.671, 99
1999	89.322, 64	102.512, 30	−13.189, 67
1998	74.705, 35	160.876, 58	−86.171, 23
1997	152.194, 91	145.383, 60	6.811, 30
1996	59.930, 53	73.512, 13	−13.581, 60
1995	102.627, 37	77.418, 48	25.208, 89
1994	85.122, 79	10.121, 73	75.001, 06
1993	125.442, 19	71.436, 86	54.005, 33
1992	107.311, 36	70.138, 88	37.172, 48
1991	103.987, 25	52.495, 63	51.491, 62
1990	26.046, 15	8.949, 50	17.096, 65
1989	??	??	3.545, 07

Bei den Ausgaben ist der Unterschied zwischen »Plan« und »Ist« mit 15000 € noch höher als bei den Einnahmen, wir haben also wesentlich weniger ausgegeben, als geplant gewesen war. Zum einen wurden von der polnischen Benutzergruppe

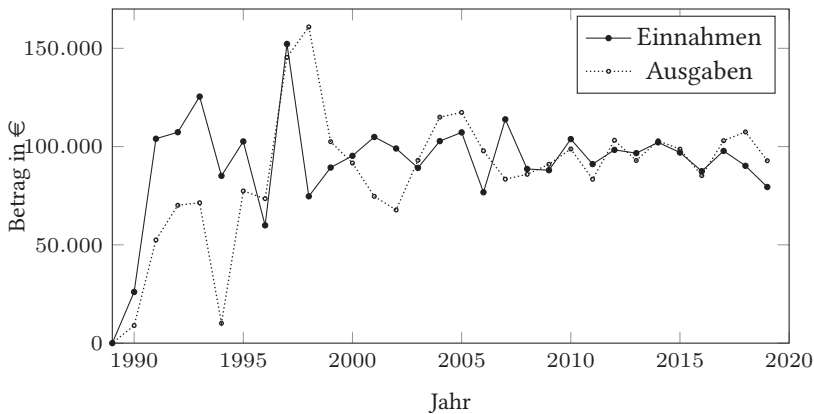


Abb. 4: Einnahmen und Ausgaben seit Vereinsgründung

GUST keine Projektmittel abgerufen, zum anderen sind im Büro weniger Kosten angefallen.

Tab. 6: Finanzplanung für 2019: Realitätsabgleich

Einnahmen			Ausgaben		
	(Plan)	(Ist)		(Plan)	(Ist)
Beiträge	75.360	70.037, 61	DVD	−1.600	−1.707, 77
TUG Kombi.	2.000	1.947, 00	TUG Kombimitgliedschaft	−2.000	−2.009, 26
Spenden	2.500	3.355, 95	DTK	−19.000	−19.584, 86
Zinsen	5	3, 78	Büro (Miete, Personal)	−40.000	−33.793, 46
Bücherverkauf	5.000	2.198, 85	Vorstand	−15.000	−13.690, 02
Sonstiges	1.000	1.879, 44	Inventar	−1000	−890, 17
			Porto	−2.500	−1.003, 99
			Web-Server	−1.800	−1.717, 26
			Förd. internationaler Tagungen	−2.000	−1.940, 53
			Open Source Veranstaltungen	−2.500	−2.890, 37
			sonstige Ausgaben	−4.000	−2.647, 02
			DANTE-Tagungen	−3.000	−2.650, 57
			Projektfonds	−7.000	−3.299, 79
			Versicherung	−500	−481, 96
			Hackercamp	−3000	−3.802, 39
			Bücher	−3.000	−720, 53
Summe	85.865, 00	79.422, 63	Summe	−107.900, 00	−92.829, 95
Unterschied	6.442, 37		Unterschied	−15.070, 05	
(Plan - Ist)			(Plan - Ist)		

Projektförderung

Der Tabelle 7 kann man entnehmen, welche Projekte wir im Finanzjahr 2019 gefördert haben. Der (erstmalige) Zuschuss zum ConT_EXt-Meeting wurde 2017 fälschlicherweise als Projekt verbucht, 2018 wurde er dann korrekt auf das Konto 443 gebucht (»Bursary für andere Tagungen«). Leider hat unsere Sekretärin auch im Jahr 2019 versehentlich wieder das Projektkonto genommen und ich habe es zu spät gemerkt.

Tab. 7: Projektförderung

gefördertes Projekt	2019	2018	2017
Serverkosten Stefan Kottwitz	1500	1500	1500
GUST Fonts	0	7000	0
ISO-Standards Ulrike Fischer		≈ 200	
L ^A T _E X-Team Meeting	≈ 500		
BayT _E X	300	≈ 190	
ConT _E Xt Meeting	1000	andere Kostenstelle 443 1000	1150
Summe	3.300	8.890	2.650

Bücher für Mitglieder

Mitgliederzahlen

Die Tabelle 9 auf Seite 20 enthält die Mitgliederzahlen aufgeschlüsselt nach Mitgliedsart ab 2007.

Für die Abbildung habe ich, falls vorhanden, Daten aus dem ersten Quartal des jeweiligen Jahres gewählt, andernfalls diejenigen Daten, die ich in den alten Ausgaben der DTK finden konnte.

Was man in der Abbildung 5 auf der nächsten Seite auch gut sehen kann, ist, dass der Mitgliedschwund der letzten Jahre zwar vorhanden, aber moderat ist.

Tab. 8: Buchver- und -einkauf

<i>Jahr</i>	<i>Verkauf/€</i>	<i>Einkauf/€</i>	<i>Saldo</i>
2007	2737	−2400	337
2008	3655	−2736	919
2009	7426	3231	4195
2010	4944	−2593	2351
2011	4274	−1484	2790
2012	6894	−3068	3826
2013	4788	−1789	2999
2014	2864	−1416	1448
2015	2667	−784	1882
2016	3960	−1875	2086
2017	4147	−2868	1279
2018	5398	−2706	2692
2019	2199	−721	1478

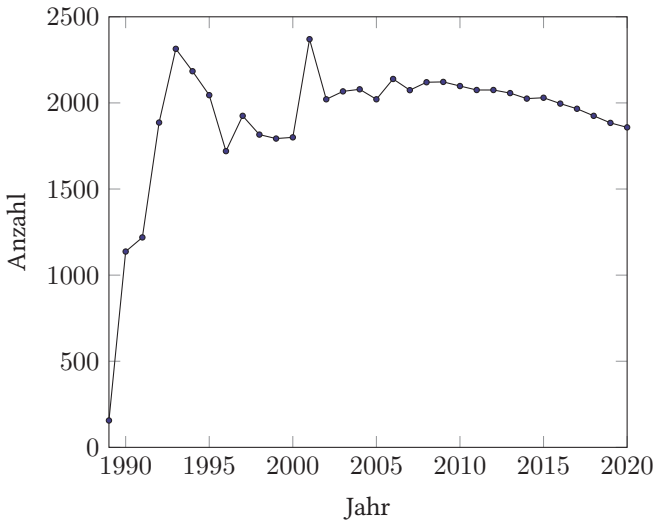


Abb. 5: Mitgliederzahlen seit Vereinsgründung

Tab. 9: Entwicklung Mitgliederzahlen (Stand jeweils DTK Nr. 1)

<i>Jahr</i>	<i>Gesamt</i>	<i>Privat</i>	<i>Institut</i>	<i>Firma</i>	<i>Ehren</i>	<i>Schnupper</i>	<i>Schüler</i>	<i>Arbeitslos</i>	<i>Rentner</i>	<i>Student</i>
2007	2074	1626	113	25	8	11	8	12	71	200
2008	2120	1674	109	26	8	11	6	11	78	197
2009	2122	1696	108	29	8	21	4	11	80	165
2010	2098	1684	107	26	8	18	9	12	77	157
2011	2075	1673	106	26	8	14	9	12	79	148
2012	2075	1681	107	27	8	15	10	11	82	134
2013	2057	1680	99	27	7	11	7	11	89	126
2014	2025	1680	93	27	7	7	3	8	97	103
2015	2030	1686	90	27	9	9	4	10	99	96
2016	1996	1652	88	28	8	8	3	9	103	97
2017	1966	1638	83	26	8	3	2	8	111	87
2018	1925	1619	77	26	8	3	3	6	114	69
2019	1884	1589	76	25	8	6	1	6	111	62
2020	1858	1564	72	26	8	5	0	7	109	67

Die Mitgliederzahlen aufgeschlüsselt nach Mitgliedsart werden in Abbildung 6 auf der nächsten Seite dargestellt, jedoch nur ab 2007.

Coronabereinigte, vorläufige (da nicht durch eine MV beschlossene) Finanzplanung für das Vereinsjahr 2020

Zur Finanzplanung gibt es anzumerken, dass

- ich die Rechnungen für 2021 nicht schon Ende 2020 verschicken werde, so dass der Betrag von 75000 € recht realistisch die Mitgliedsbeiträge für das Vereinsjahr 2020 angibt,
- die Vorstandsspesen heuer nicht so hoch ausfallen werden, da keine internationalen Tagungen mit Langstreckenflügen finanziert werden müssen. Da bereits die Frühjahrstagung in Deutschland weggefallen ist und die Herbsttagung wahrscheinlich nicht »vor Ort« stattfinden wird, ist es gut möglich, dass hier wesentlich geringere Kosten, als geplant, anfallen werden;
- wir zwar im Jahr 2019 für Büromiete, -nebenkosten und -personal nur knapp 35000 € aufbringen mussten, dass sich das aber heuer noch ändern kann, da wir eine Nachfolge für unsere Sekretärin suchen müssen.
- der Posten für die Tagungen nur deshalb nicht auf 0 € angesetzt wurde, weil wir eventuell Kosten für die digitale Variante stemmen müssen,
- wohl kein größerer Betrag aus dem Projektfonds entnommen werden wird, da seitens der GUST seit 2018 kein Antrag auf Förderung mehr gestellt wurde. Der von der 56. MV in Zeuthen im Jahr 2017 um 20000 € aufgestockte Projektfonds sollte eigentlich jährlich 7000 € für die GUST bereitstellen.

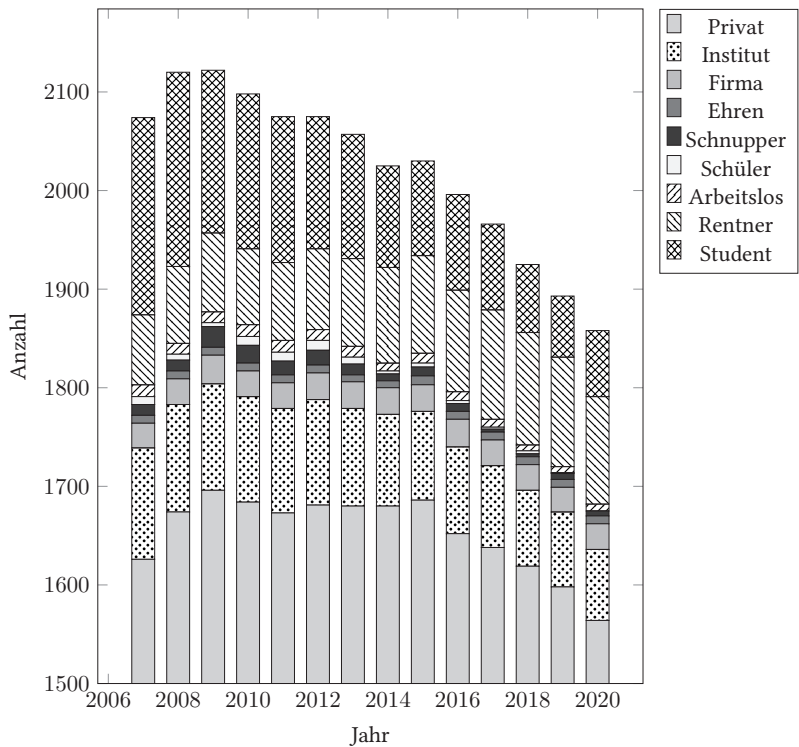


Abb. 6: Zusammensetzung der Mitglieder

Tab. 10: Finanzplanung für 2020

<i>Einnahmen</i>	<i>Betrag</i>	<i>Ausgaben</i>	<i>Betrag</i>
Beiträge (ohne TUG)	75.000	DVD	−1.700
TUG	2.000	TUG	−2.000
Spenden	2.500	DTK	−20.000
Zinsen	5	Büro (Miete, Personal)	−40.000
Bücherverkauf	2.000	Vorstand	−8.000
Sonstiges	1.000	Inventar	−3.000
		Porto	−2.500
		Web-Server	−1.800
		Förd. internationaler Tagungen	−2.000
		Open Source Veranstaltungen	−500
		sonstige Ausgaben	−4.000
		DANTE-Tagungen	−1.000
		Projektfonds	−3.000
		Versicherung	−500
		Bücher	−3.000
Summe	82.505	Summe	−93.000
		Saldo	−10.495

CCCamp19

Doris Behrendt, Mario Haustein, Johannes Hielscher, Nils Pickert, Henning Hraban Ramm

Liebe Mitglieder, es hat wieder eine Weile gedauert, aber hier ist er nun: der Bericht zur Dantepräsenz beim Chaos Communication Camp 2019 des Chaos Computer Clubs CCC im Ziegeleipark Mildenberg.

Die Bewohner des L^AT_EX-Village als Vertreter von DANTE e.V. waren neben mir (Doris) Johannes, Mario und Nils, außerdem für die ConTeXt Group Hraban. Bei allen vier möchte ich mich an dieser Stelle im Namen von Dante e.V. ganz herzlich bedanken. Auch Marei, die nicht im L^AT_EX-Village wohnte, da sie ihren »Ehrenamts-schwerpunkt« in die CCC-Camp-Orga gelegt hatte, soll nicht unerwähnt bleiben, immerhin hat sie trotz Schlafmangels beim Entladen des LKW geholfen und einen Vortrag gehalten.

Vorbereitungen

Doris Behrendt

Wer den Finanzbericht studiert, wird sehen, dass die Präsenz von Dante e.V. auf dem Camp in Form eines L^AT_EX-Village einiges kostet. Zuerst einmal hatten wir dieses Jahr ein stabiles Zelt mit Holzboden und sechs Quadratmetern Fläche »gemietet«, das man vor Ort aufgebaut vorfindet, wenn man ankommt. Das lag bei ca. 1400 €. Das Foto 1 zeigt Nils vor dem Zelt.

Im Zelt wurden die Vorträge gehalten, außerdem hatten wir dort Ansichtsexemplare unserer L^AT_EX-Bücher ausgelegt und eine Kiste mit allen Ausgaben der DTK seit Vereinsgründung hatten wir auch dabei.

Ein weiterer großer Posten von über 1000 € war die LKW-Miete, außerdem wurden allen aktiv Helfenden auf Wunsch die Kosten für die Eintrittskarten, die bei ca. 300 € lagen, erstattet.

Schon am Freitag vorher wurde bei mir zu Hause der LKW beladen; das Foto 2 zeigt nur einen Teil der Dinge, die hinein mussten: Projektionsleinwand für den Beamer, Beamer, Kabel Kabel Kabel, Werkzeug, Trampolin für die Kinder, Goldfolie als Klimaanlage und Sonnenschutz, denn das Vortragszelt muss möglichst gut verdunkelt werden, sonst kann man die BeamerSlides nicht lesen. Küchenequipment, Milchkanne für Wasser, Käscher für den Pool. Der »Running Gag« auf den Camps ist ja der, dass immer von offizieller Seite angesagt wird, dass man auf keinen Fall



Abb. 1: Nils Pickert

Pools aufstellen soll. Wenn man aber schon öfter auf dem Camp war, weiß man, dass man natürlich erst recht einen mitzubringen hat, damit die Anzahl der Pools in der Camp-Abschlussstatistik möglichst hoch ausfällt. Wir erfreuen uns immer an unserem Pool, schließlich muss man irgendwie das Bier kühlen während der Zeit, in der es noch keinen Strom für den Kühlschrank¹ im Zelt gibt. Wenn man zu denen gehört, die schon einige Tage vor dem offiziellen Beginn anreisen, muss man eine Weile ohne Strom und Internet auskommen und dafür vorsorgen. Ein ebenfalls systemrelevanter Ausrüstungsgegenstand ist immer wieder unser »Milchwägle«, mit dem man zur Not sechs volle Milchkannen oder alternativ sechs Kubikmeter Zeltlagerausrüstung über weite holprige Strecken mit minimalem Kraftaufwand transportieren kann.

Das Mobiliar für das Zelt und den Außenbereich davor bestand aus Biergarnituren und einer kleinen Sitzcke aus bepolsterten Europaletten, die wir in der Nähe meines Wohnortes gemietet haben und im LKW nach Berlin gefahren haben, das war wesentlich günstiger, als die Biergarnituren vor Ort zu leihen.

Das Foto 3 zeigt Mario Haustein vor unserem Sitzbereich.

Die Kernzeit des Camps ging von Mittwoch bis Sonntag, das Vortragsprogramm kann man sich hier <https://fahrplan.events.ccc.de/camp/2019/Fahrplan/index>.

¹ »Natürlich« hatten wir einen Campkühlschrank und gutes Fränkisches Meister-Bier dabei.



Abb. 2: Scheune mit einem Teil der LKW-Ladung



Abb. 3: Mario Haustein

html ansehen. Die meisten Vorträge gibt es als Film auf <https://media.ccc.de>. Der Vortrag von Marei Peischl beispielsweise steht im »Fahrplan« hier: <https://fahrplan.events.ccc.de/camp/2019/Fahrplan/events/10223.html> Als Video findet man ihn hier: https://media.ccc.de/v/Camp2019-10223-schleuderpackung_2_0



Die Vorträge und Workshops, die wir im LaTeX-Village angeboten haben, waren im Wiki zu finden: <https://events.ccc.de/camp/2019/wiki/Village:LaTeX>

Da man ab Montag schon nicht mehr mit dem LKW auf das Gelände fahren durfte, haben wir am Freitag vorher beladen und am Samstag die Fahrt von Würzburg nach Berlin hinter uns gebracht.

Aus meiner subjektiven Sicht in der Tat »hinter mich gebracht«, denn ich litt unter einem akuten Bandscheibenvorfall, wollte das aber irgendwie nicht richtig wahr haben und lieber verdrängen und das LaTeX-Village »durchziehen«, deshalb bin ich erst nach dem Camp zum Arzt gegangen. Beziehungsweise, das stimmt nicht, die Schmerzen wurden nach ein paar Tagen so schlimm, dass ich am Tag 2 das Camplazarett aufsuchte. Dort wurde ich sehr kompetent von *Mad Doc* behandelt, ein Orthopäde, der mir viel Ibuprofen und ein Magenmittel verschrieb und mir zudem eine Wärmflasche (bei der Hitze!) sowie die horizontale Lebenslage empfahl. Und so verbrachte ich leider 95 % der Campzeit so, wie die Leute auf dem Zauberberg.

Mittwoch – Tag 1

Nils Pickert

Der erste offizielle Tag des Camps war am Mittwoch, inklusive einer Eröffnungszeremonie. Nachdem ich mich schon fast an das Übernachten im Zelt und den wenigen Schlaf gewöhnt hatte, ließ ich mir jedoch diese Eröffnungszeremonie entgehen und besuchte den Hamburger CCC zu einer Einführung in die Bedienung eines Strüver Schienenkulis. Am Abend vorher bin ich mit einem der Hamburger ins Gespräch gekommen, als dieser mit besagtem Schienenkuli – einer kleinen Feldbahnlokomotive (siehe Foto 9) – Probefahrten auf den Gleisen des Ziegeleimuseums machte. Wie sich herausstellte, wurden noch weitere Hilfslokführer gesucht, um in den Tagen des Camps regelmäßigen Zugverkehr zu veranstalten.

Nach einer kurzen Einweisung und einigen Probefahrten wurde ich dann als Train-Angel (oder auch *Traingel*) ins Schicht-System aufgenommen und fuhr in den folgenden Tagen mit Kindern und Erwachsenen in den Passagierloren fröhlich um das Camp-Gelände.

Nach dem Mittagessen ging es dann mit unseren angebotenen Kursen los. Hraban startete mit der ConTeXt-Einführung, bei der ich dann auch erste Schritte mit dieser



Abb. 4: Der Löwe hat sein Zelt bezogen.

Variante der $\text{\TeX}$ -Familie machte. Fazit: schaut interessant aus, muss ich mal mehr mit herumspielen.

Anschließend kam dann meine Einführung in $\text{\LaTeX}$ für absolute Anfänger, die auch einige Besucher hatte. In den folgenden Tagen kamen dann tatsächlich noch mehr... Obwohl auf dem Camp überwiegend Leute unterwegs sind, die Kommandozeilen gewöhnt sind und in abstrusesten Programmiersprachen arbeiten, gab es doch einige, die den Erstkontakt mit $\text{\TeX}$ in unseren Einführungen hatten oder deren $\text{\TeX}$ -Erfahrungen so lange zurück lagen, dass sie doch noch einmal bei einer Einführung dabei sein wollten.

Abends war ich dann wieder als Feldbahner unterwegs, nach der Schicht bewunderte ich die vielen blinkenden Dinge überall. Das gesamte Camp-Gelände war eine einzige große Kunst-Installation und zu jedem blinkenden Ding gab es Leute, die einem gerne dazu etwas erzählten oder sogar Workshops zum selber Löten angeboten haben.

Fazit nach dem ersten Tag: Schon am ersten Tag viele interessante Leute die spannende Sachen machen, sehr coole Ideen überall und vor allem ist es immer wieder beeindruckend, wie eine solche Riesen-Veranstaltung von Freiwilligen organisiert

und betrieben wird. Wie auch auf dem CCC-Kongress hatte ich nach dem ersten Tag bereits das Gefühl, dass das Hirn voll ist und betrat den Zwischenraum zwischen Reizüberflutung und Faszination, was alles möglich ist.

Donnerstag – Tag 2

Henning Hraban Ramm

Nach einer wie üblich kurzen Nacht begann Tag 1 mit dem Anstehen an der Dusche und Wäschewaschen. Das Wetter war ideal – sonnig und trocken, nicht zu heiß, aber warm genug für meinen Sohn, längere Zeit im Badeteich zu verbringen.

Meine Kinder (7 und 9 Jahre alt) löteten in fast vier Stunden konzentrierter Arbeit einen Bakiwi-Laufroboter zusammen, der abends die ersten Schritte auf wackeligen Drahtbeinen machte. (Und leider am nächsten Tag schon wieder kaputt war, weil der verwendete Heißkleber offenbar nicht für die Kunststoffteile geeignet war.)

Die Feldbahn brachte uns vom RobotVillage mit großzügigen Umwegen zurück nach Hause; ich schlug vor, die kleine Diesellok »Rattelschneck« zu taufen, aber die Betreiber wollten doch lieber bei der Typenbezeichnung »Schienenkuli« bleiben.



Abb. 5: Auf Chaos-Veranstaltung gibt es nur bunte Beleuchtungen.

Zwischendurch hielt ich den zweiten Durchgang meiner Con_TE_Xt-Einführung vor etwa zehn interessierten Teilnehmern, diesmal auf englisch und schon deutlich strukturierter als an Tag 0, aber immer noch mit Potential für Verbesserungen.

Nach mir war wiederum Nils mit seiner bewährten L^AT_EX-Einführung an der Reihe, der ich allerdings kein zweites Mal beiwohnte. Unser Zeitplan (je eine Stunde ohne Pause) hatte sich schon am Mittwoch als zu knapp erwiesen, jedoch wollten wir die angekündigten Zeiten nicht mehr ändern. Doris referierte später souverän trotz heftiger Schmerzen über Grafiken mit L^AT_EX.

Auf dem ganzen Camp habe ich außer den Workshops des Makerspace X-Hain, an denen ich mit meinen Kindern teilnahm, keine »fremden« Veranstaltungen mitbekommen, dafür hatte ich einfach keine Kapazitäten frei. Mit meinen aktuellen T_EX-Projekten voran zu kommen, war ohnehin arg optimistisch; auf einem Festival wie dem CCCamp sollte man nicht mit der produktiven Atmosphäre der Con_TE_Xt-Meetings rechnen.

Immerhin haben Nils und ich uns einen Wettbewerb geliefert, das Beschwerdeformular der Chaospost neu zu setzen, nachdem Mario die Beschwerde eingereicht hatte, dass es nicht mit L^AT_EX gesetzt sei. Ergebnis: Ich war mit Con_TE_Xt schneller fertig (komplexe Tabellen sind mit »natural tables« kein Problem), aber Nils' L^AT_EX-Version (L^AT_EX 3 mit »coffins«) war durch die Verwendung eines Signaturfeldes vollständig. Meine Anfälligkeit für autistische Overloads wurde durch den chronischen Schlafmangel verstärkt – Lärm² bis vier Uhr nachts und Kinder, die dessen ungeachtet um sieben Uhr aufstehen, lassen die Nächte arg kurz werden.

Fazit: Beim nächsten Lager bin ich wieder dabei, werde mir aber eine ruhige Ecke zum Schlafen suchen.

Samstag – Tag 3

Mario Haustein

Tag 3 begann für mich um 0:00 Uhr. Zu diesem Zeitpunkt hatte ich mich für zwei aufeinanderfolgende zweistündige Engelschichten an der Einlasskontrolle gemeldet. Engel nennt man die freiwilligen Helfer, die die Veranstaltungen am laufen halten. Für anstehende Aufgaben kann man sich über ein Schichtplanungssystem eintragen. Das Betätigungsfeld ist vielfältig und reicht von einfachen Handlangertätigkeiten über Küchenarbeit, der technischen Betreuung der Camp-Infrastruktur (Datennetz, Aufzeichnungs- und Übertragungstechnik, ...) bis hin zum Sanitätsdienst. Ich beschränkte mich auf einfache Aufgaben, wie das Aufstellen von Zäunen und Stühlen,

² Es ist schon erstaunlich, was manche Lagerteilnehmer glauben, ihren Mitmenschen zumuten zu dürfen – eine Diskussion über Musikgeschmack will ich gar nicht anfangen.



Abb. 6: Hier waren Künstler am Werk. Nur bei Nacht sieht man, dass eine Discokugel mitten im Wald hängt.

sowie die Nachtwache in den Vortragszelten und die Einlasskontrolle. Das gesamte L^AT_EX-Village stand dem nicht nach. Mehr als Hälfte von uns hat sich als Engel eingebracht: egal ob als einfacher Helfer, Lokführer für die Feldbahn, Sanitäter oder als Ansprechpartner für Autisten.

Sinn und Zweck der Engelschichten ist es auch, ins Gespräch mit anderen Campteilnehmern zu kommen. Schnell erfährt man aus welchem Village die neu gewonnenen Bekannten stammen. Die Reaktionen waren meist die gleichen, wenn ich offenbarte, dass ich aus dem L^AT_EX-Village stamme: »Oh, L^AT_EX – damit habe ich meine Abschlussarbeit geschrieben.« in einigen Fällen ergänzt durch »Aber mein jetziger Arbeitgeber verlangt von mir, dass ich eine andere Software verwende.«

Nach dem Ende meiner zweiten Schicht um 4:00 morgens bot sich für mich die Gelegenheit, auf dem Camp-Gelände fotografieren zu können. Um diese Zeit schlafen die meisten Hacker. Die Fotorichtlinie des CCC verlangt, dass man aus Gründen des Persönlichkeitsschutzes zunächst alle Personen auf einem Bild um Erlaubnis fragen muss, *bevor* man das Foto macht. Zu normalen Zeiten ist das eine praktisch unlösbare Aufgabe – und der Grund warum das Camp auf Bildern so menschenleer aussieht.

Nun war es aber an der Zeit ins Bett zu gehen. Für den Rest des Tages hatte ich mir vorgenommen, das Camp in Ruhe zu erkunden und für diesen Tag keine weiteren Engelschichten zu absolvieren. Wie auf einem Hackercamp üblich, haben die Menschen vor Ort mehr oder weniger exotische Hobbies. Wie sich durch Zufall herausstellte, ist Martin aus unserem Village ebenfalls lizenzierter Funkamateur. Wir entschlossen uns, gemeinsam am Fuchsjagdparkur der Chaoswelle teilzunehmen. Dabei sucht man mit einer Richtantenne und einem Empfänger die sog. Füchse – kleine, im Gelände versteckte Sender mit geringer Reichweite. Bei jedem Sender befand sich ein Code, den man im Anschluss als Nachweis in das Auswerteformular eingeben musste. Leider stellte das Formular kein Eingabefeld für den Entdeckungszeitpunkt zur Verfügung. Da ich alle gefundenen Sender im Anschluss gesammelt meldete, führte mich das ungewollt mit einer Gesamtzeit von reichlich 2 Minuten an den Anfang der Rangliste.

Einer der vielen Running Gags auf dem Camp war das Salzamt, welches als offizielle Beschwerdestelle fungierte.³ Beschwerden waren über ein offizielles Formular einzureichen. Dass das Formular nicht mit $\text{\TeX}$ erstellt wurde, konnten wir nicht hinnehmen und veranlasste uns prompt zu einer Beschwerde. Für eine ordnungsgemäße Bearbeitung war es allerdings notwendig, das Formular mit dem offiziellen Dienstsiegel des Salzamts abstempeln zu lassen. Wie es sich für eine Behörde gehört, war der entsprechende Schalter des Amts jedoch nie besetzt. Mit Hilfe der Chaospost konnten wir jedoch einen Stempel ergattern und die Bearbeitungsnummer »CP-002« legt nahe, dass dies noch nicht viele Personen vor uns geschafft hatten. Dass der Einwurfschlitz des Beschwerdebriefkastens das vertraute Geräusch eines Aktenvernichters von sich gab, verunsicherte uns letztendlich aber sehr, ob unsere Beschwerde tatsächlich ernst genommen wurde.

Auch beim Camp hat sich gezeigt, dass Chaos-Veranstaltungen leider viel zu kurz sind, um an allen interessanten Aktivitäten teilnehmen zu können. Eines steht für mich fest. Das Chaos Communication Camp war mein erstes, aber nicht das letzte Camp, welches ich besucht habe.

Samstag – Tag 4

Johannes Hielscher

Der Tag, an dem wir den Lagerkoller überlistet haben ...

Auf dem Programm des $\text{\LaTeX}$ -Zelts standen wieder Nils (*How to set up a Gitlab project to automatically generate a Website and PDFs from the same source and make normal people use it...*) und Doris zum Thema *Grafiken mit $\text{\TeX}$ anlegen*. Wieder erreichten

³ Der Wikipedia-Artikel fasst den Sinn und Zweck dieser nicht existierenden Behörde recht treffend zusammen: <https://de.wikipedia.org/wiki/Salzamt>

Nach Bearbeitung einschicken an: (Name, Name w. V. der Chaos-Post verschicken)

NACHNAME	VORNAME	SEITE	Landes-Post
hamari		2426	170

Chaos-Post Beschwerdeformular

Informationen für die Salzzamt:

NACHNAME (in Blockbuchst.)
HAMARI

E-Mail-Adresse
[redacted] (Zusatz)

Erster vollständiger Nachname (alternativ erste verwendete E-Mail-Adresse)
hamari

GEBURTSDATUM [redacted] TYP Hackerfräule ☒ Hacker ☐ RELIGION keine

Ein KREIS (gemäß aktuellen Verordnungen, bei Nichtzustand des Kreises mit der geringsten Schätzung vor sich bringen)

Experte für:
☐ Art ☐ Blockaden ☐ Instruktion & D. ☐ Kundenbegegnung ☐ SCDA (mit Aufzeichnung) oder sonst
☐ Das menschlich-akustische Komplex ☐ Programmieren (HTML) ☐ Programmieren (JavaScript) ☒ Cyber ☐ Ich so

STAATSGEBHÖRIGKEIT
 DEUTSCHLAND ☒ Deutschsprachig ☐ sonstige ☐

Angabe der IDG-Nummer (sofern bekannt): [redacted]

BESCHWERDENUMMER (vom bearbeitenden Beamten ausfüllen)
CP-002

BESCHWERDE über: Adresse (Nachname oder E-Mail-Adresse) **Ersteller des Formulars** Priorität **0 10**

Art der Beschwerde
Verhärzung

Ist diese Beschwerde dringend? ja ☒ nein ☐

Sei Anhang  **Dieses Formular sollte in LaTeX gesetzt werden**

Wurden diese Angaben mit dem Adressaten abgesprochen?
 ja ☐ nein ☒

Beschwerde über das Salzamt (Bitte dafür diesen Bereich verwenden)
 Zweigstelle [redacted] Priorität [redacted]

An der Beschwerde [redacted]

Ist diese Beschwerde dringend? ja ☐ nein ☒ **dringend**

Unterschrift (Sonst: Beschwerdeführer's/Beschwerdeführer's)
hamari

Arbeitsstempel des Salzamts (Beschwerde) (Bitte nicht unterschreiben)



Abb. 7: Unsere Beschwerde beim Salzamt

sie mit ihren, jeweils bis ans Limit der Sitzplätze nachgefragten Workshops, ein interessiertes und motiviertes Publikum von anderen Camp-Teilnehmern.

Ein weiteres Mal kam an diesem Tag $\text{\TeX}$ an prominenter Stelle zur Sprache: Auf der Bühne eines der großen Vortrags-Zelte erzählte Marei Peischl von der Vereinszeitschrift *Datenschleuder* des CCC: Sie hat kräftig mit angepackt, dieses Organ, nach längerer Pause, wiederzubeleben. Das Heft kommt nun in einem schicken, durchaus an die $\text{\TeX}$ nische Komödie erinnernden Layout daher – gesetzt mit $\text{\LaTeX}$. Zu dem Apparat an Helferlein im Hintergrund wusste Marei einiges zu erzählen: »Häcksereien«, um die Inhalte $\text{\TeX}$ -konform zu bekommen (inkl. Autorentisziplin), hexadezimale Seitenzahlen, Barrierefreiheit (mit ePUB-Export), Registerhaltigkeit ...

Insgesamt hatte auch der vorletzte Programm-Tag des Camps ein Programm zu bieten, bei dem jeder neugierige Besucher bedauern muss, nicht überall gleichzeitig zu sein. Dennoch stellten wir uns schon langsam auf das Abbauen ein (Zelt aufräumen, Resteessen). Ringsum war abends noch volle Partystimmung, doch unser *Loewe Opta*-Röhrenradio hat das Fass zum Überlaufen gebracht. Nach professionellen Verkabelungsversuchen als Endstufe einer Digitalanlage doch auf gutes altes UKW eingestellt, war der 50er-Mono-Sound anderen Gästen zu schrill. Und so begnügten wir uns mit der Retro-Optik im Schein der allgegenwärtigen LEDs.

Sonntag – Tag 5

Doris Behrendt

Da ich, wie eingangs schon erwähnt, nicht so viel Berichtenswertes erlebt habe, müssen Sie auf einen Bericht zu Tag 5 verzichten, stattdessen dürfen Sie sich noch einige Fotos anschauen. Außerdem hier noch der Link zum Code für das Beschwerdeformular von Nils <https://gitlab.com/nilspickert/chaospost-beschwerdeformular>. Den Con $\text{\TeX}$ t-Code von Hraban kann man auf der Homepage von DANTE e.V. herunterladen: <https://www.dante.de/veranstaltungen/dante-unterwegs/>



Das Foto 10 zeigt einen der Stromgeneratoren, die während der Kerncampzeit die Stromversorgung sicherstellten. Für diejenigen, deren »schade-dass-es-vorbei-ist-Frust« nach dem Camp zu groß wurde, gab es das Generatorgeräusch auf YouTube: https://www.youtube.com/watch?v=vUQ-m_S4es4&feature=youtu.be

Die Abbildung 11 zeigt unser Vortragszelt von innen, wo Hraban gerade einen Vortrag hält.

Das Bild 12 zeigt noch einmal alle Villagebewohner, unter ihnen waren auch einige Nicht-Dantemitglieder. Vorne von links nach rechts Nils, Mario, Sabine (Doris' Tochter), Merle und Lenz (Hrabans Kinder), Wolfgang (Doris' Sohn), Doris. Zweite

Reihe von links nach rechts Paula (schlecht zu sehen, Beates Tochter), Johannes, Martin (Doris' Bruder, der mit dem Röhrenradio, zuständig für Elektrotechnik und -installation), Beate (Doris' Freundin, zuständig für die Versorgung mit Whiskey), Hraban, Latschi (Doris' Kumpel, zuständig für Kinderbespaßung). Nicht im Bild Doris' Mann, der das Foto aufgenommen hat. Er war viel mit Kochen beschäftigt und hat den LKW gefahren. Obwohl für die Rückfahrt des LKW eigentlich Johannes eingeplant gewesen war. Der hatte mir allerdings nicht gesagt, dass er gar keinen Führerschein hat, und ich war überhaupt nicht auf die Idee gekommen, dass ein erwachsener Mann heutzutage keinen Führerschein haben könnte. Na gut, und so hat auch bei diesem Camp mein Mann die organisatorischen Löcher gestopft, sogar als Sanitätsengel hat er sich dieses Mal verdingt und Nachtschicht geschoben: Danke nochmal an alle, die dabei gewesen sind und mit angefasst haben!



Abb. 8: Keine Nacht verging ohne Light- und Lasershows.



Abb. 9: Feldbahnlokomotive



Abb. 10: Bredenoord



Abb. 11: Hraban beim Vortrag



Abb. 12: Alle Villagebewohner

Bretter, die die Welt bedeuten

n-doc – ein L^AT_EX-basiertes Verfahren für IT-Sicherheitszertifizierungen

Alexander Krumeich

Geräte und Software im Umfeld der Telematikinfrastruktur des Gesundheitswesens (TI) müssen hohe Sicherheitsstandards erfüllen. Im Falle des eHealth-Konnektors zum Anschluss der Arztpraxen und Krankenhäuser an die TI bedingt dies eine Zertifizierung nach *Common Criteria* (CC) durch das Bundesamt für Sicherheit in der Informationstechnik (BSI).¹

In einem Zertifizierungsverfahren nach CC dokumentiert der Hersteller die Sicherheitseigenschaften des zu evaluierenden Gegenstandes. Diese Dokumente können umfangreich und komplex werden. Ein solches Zertifizierungsverfahren lässt sich nur schwer mit den klassischen Office-Werkzeugen bearbeiten. Dennoch sind diese Werkzeuge in den Unternehmen häufig das (einzige) Mittel der Wahl. Wir haben stattdessen den Versuch gewagt, L^AT_EX als Dokumentationswerkzeug auszuprobieren. Im Laufe von drei Jahren ist dabei aus einem kleinen Experiment eine ausgewachsene Dokumentationsplattform geworden, die wir Anfang 2020 als Open-Source-Software veröffentlicht haben. [4] In diesem Artikel beschreiben wir die Besonderheiten unseres Verfahrens und gehen auf die Implementierung ein.

Problematik und Anforderungen

Für die Zertifizierung unseres Konnektors müssen wir als Hersteller ein gutes Dutzend Dokumente unterschiedlichen Umfangs an das Prüflabor liefern: von ca. 30 Seiten Fließtext bis hin zu Testdokumentationen mit knapp 3000 Seiten teilweise generierten Textes.

Technische Dokumentation ist darauf angewiesen, inhaltlich und formal konsistent zu sein: Gleiche Dinge müssen immer gleich heißen und typographisch eindeutig

¹ Die Common Criteria sind ein internationaler Standard (ISO/IEC 15408) für die Dokumentation und Bewertung der Sicherheitseigenschaften von IT-Produkten. [7] Das Rahmenwerk beschreibt Dokumenttypen und Prüfmethode, die Herstellern und Prüflaboren ein vergleichbares, technologie-neutrales Vorgehen ermöglichen.

gekennzeichnet sein. Weiterhin werden diese Dokumente selten linear gelesen. Es ist also notwendig, eine non-lineare Navigation (über Hyperlinks) zu ermöglichen. Allerdings sollen die Autorinnen und Autoren nicht zuviel Arbeit in den Verwaltungsaufwand für die Navigation stecken: Die Hypertextstruktur soll möglichst automatisch generiert werden.

Außerdem gibt es die organisatorische Notwendigkeit, dass mehrere Personen simultan an den Dokumenten arbeiten sollen, ohne sich dabei in die Quere zu kommen. Die Dokumente müssen dafür stark modularisiert sein und sich leicht und schnell zu größeren Einheiten integrieren lassen. Gleichzeitig ist es notwendig, die Dokumente unter Versionskontrolle zu haben, um verschiedene Bearbeitungs- und Releasestände zuverlässig wiederherstellen zu können.

Nicht zu vergessen ist die Akzeptanz der Autorinnen und Autoren. Dokumentationen zu schreiben, macht selten Spaß, wenn man eigentlich Software entwickeln möchte. Hier wird ein System benötigt, dass die spezifische Denk- und Arbeitsweise von Softwareentwicklerinnen und -entwicklern unterstützt.

Eine besondere Herausforderung ist es, die vielen formalen Elemente der CC (wie *Bedrohungen*, *Sicherheitsziele*, *Sicherheitsanforderungen*) auf die Eigenschaften des Evaluierungsgegenstandes (u. a. *Schnittstellen*, *Subsysteme*, *Module*) abzubilden: Welches Modul setzt welche Sicherheitsanforderung um? Welche Sicherheitsfunktionalität ist über welche Schnittstelle erreichbar?

Schließlich muss auch gewährleistet sein, dass die Dokumente regelmäßig neu erzeugt und zuverlässig an leicht erreichbaren Orten abgelegt werden. Der Releaseprozess muss nachvollziehbar sein und die ausgelieferten Dokumente müssen archiviert werden.

Lösung: L^AT_EX statt Office

Die meisten dieser Punkte werden durch L^AT_EX erfüllt: Große Dokumente sind kein Problem. Durch zentral erstellte Makros muss sich keine Autorin und kein Autor eigene Gedanken über Formatierungen machen. Modularisierung ist über `\input`-Anweisungen leicht zu realisieren. Die Quelldateien sind reiner Text in UTF-8 und lassen sich hervorragend in Versionskontrollsystemen (hier: Git) verwalten. Der Workflow des Schreibens, Übersetzens, Betrachtens ist sehr nah an dem der Softwareentwicklung. Für die Dinge, die L^AT_EX nicht von Hause aus mitbringt, haben wir eigene Funktionen entwickelt. Dabei haben wir auf die Möglichkeiten zurückgegriffen, die LuaT_EX bietet.

Domänenspezifische Makros

Zu Beginn der Überlegungen von einem Office-Produkt auf L^AT_EX umzusteigen, stand der Wunsch nach einfacher und konsistenter typographischer Auszeichnung der Texte. Das Formatieren in einem WYSIWYG-Programm ist auf den ersten Blick einfach, hat aber auch Nachteile, die für uns sehr kritisch wurden. Besonders schwierig war, dass die Verwendung von Druckformaten nicht erzwungen werden kann. Sobald mehrere Autorinnen und Autoren an den Dokumenten arbeiten, besteht das Risiko, dass Auszeichnungen und Schriftarten nach eigenem Gusto ausgewählt werden. Die Nachbearbeitung zur Vereinheitlichung ist sehr aufwändig. Bei der Umstellung auf L^AT_EX haben wir von vornherein Wert darauf gelegt, dass ausschließlich zentral erstellte Makros verwendet werden. Diese Makros sind semantisch orientiert. Ein Beispiel illustriert dies: Schlüsselworte aus der Spezifikation sollen in serifenloser Schrift und kursiv gesetzt werden. Statt direkte Formatierungen wie `\textsf{\textit{Schlüsselwort}}` zu verwenden, definieren wir das Makro `\keyword{}`, das diese Auszeichnungen mit sich bringt. Das hat den offensichtlichen Vorteil, dass sich die Formatierung zentral anpassen lässt. Viel wichtiger jedoch ist, dass die Autorinnen und Autoren nicht mehr in Kategorien der Formatierung denken, sondern sich an der Bedeutung der Makronamen orientieren.

Durch den Werkzeugkasten der zentral definierten Makros ergibt sich eine »ubiquitäre Sprache«.² Die ubiquitäre Sprache soll sicher stellen, dass gleiche Dinge von allen beteiligten Personen immer gleich benannt werden. Dies lohnt sich besonders, wenn Begriffe in mehreren Konnotationen verwendet werden: Ein Schlüsselwort aus der Spezifikation wird ebenfalls als der Name einer Konstante im Code verwendet. Hier helfen Makros, die Zuordnung eindeutig zu machen:

Tritt der Fehler `\keyword{INVALID_CREDENTIALS}` auf, wird die Exception mit dem ↪Parameter `\code{INVALID_CREDENTIALS}` erzeugt.

An diesem Beispiel ist erkennbar, dass die Verwendung von Makros mehr ist als die Umsetzung typographischer Vorgaben. Die Makros annotieren den Text und unterstützen die Autorinnen und Autoren bei der Zuordnung von Konzepten. Wir nennen dieses Vorgehen *semantisches Markup*.

Andere Begriffe werden ebenfalls durch Makros gekapselt. Auch hier ein Beispiel: der Produktname »KoCo Box MED+«. Die Kombination aus Groß- und Kleinschreibung, Leerzeichen und dem »+« am Ende lässt viel Spielraum für Varianten. Das Makro `\kocobox` stellt sicher, dass es eine einheitliche, konsistente Schreibweise gibt.

Wir haben einen Satz domänenspezifischer Makros entwickelt, wie sie in einem CC-Dokument vorkommen. Die Autorinnen und Autoren verwenden

² Dieser Begriff stammt aus dem Domain Driven Design, einer Technik in der Softwareentwicklung. [2]

ausschließlich diese Makros. So ergibt sich ein einheitliches Erscheinungsbild innerhalb eines Dokuments. Dadurch, dass die Makros in einer eigenen Datei zentral definiert und in alle Dokumente eingebunden werden, ist die Konsistenz auch dokumentübergreifend garantiert. Teilweise rufen die Makros Lua-Funktionen auf, um den übergebenen Text vorzuverarbeiten. Ein Klassiker sind die Unterstriche, die häufig verwendet werden, aber nicht die Tiefstellung im Formelmodus auslösen sollen. Unser Makro `\underscore` nutzt die Textersetzung in Lua mit einem regulären Ausdruck, um den Unterstrich zu maskieren: `\newcommand{\underscore}{\directlua{string.gsub("#1", "_", "\_")}}`. Dieses Makro wird ebenfalls von `\keyword` verwendet, sodass die Autorin und der Autor den Unterstrich verwenden kann, ohne sich über dessen Sonderbedeutung in T_EX im Klaren sein zu müssen.

Datenbankmodell des Evaluierungsgegenstandes

Eine der Herausforderungen in einem CC-Verfahren ist es, den Überblick über die Relationen zwischen den Begriffen der Common Criteria und den Eigenschaften des Evaluierungsgegenstandes zu behalten. In unserem Verfahren hat es sich sehr bewährt, dass wir diese Relationen in einer Datenbank modelliert haben. Wir nutzen die Datenbank für mehrere Zwecke, u. a. zur Auflösung von Abkürzungen oder zum Generieren von Tabellen.

Auflösen von Abkürzungen

Die formalen Sicherheitsanforderungen der CC tragen Namen, die auf den ersten Blick gewöhnungsbedürftig sind. Ein Name wie »FCS_COP.1/NK.TLS.HMAC«³ geht nicht leicht von der Hand und lädt zu Tippfehlern ein. Stattdessen verwenden die Redakteure das Makro `\sfr{fcs_cop.1/nk.tls.hmac}`.⁴ Das Argument des Makros `fcs_cop.1/nk.tls.hmac` wird als Zugriffsschlüssel für den Datenbankzugriff verwendet. Das sieht zwar nicht unbedingt einfacher aus, doch falls sich der Schlüssel nicht über die Datenbank auflösen lässt, erzeugen die Makros und Lua-Programme eine Fehlermeldung, die im Text sofort auffällt. Damit sind Abweichungen durch Tippfehler ausgeschlossen.

³ Definiert Anforderungen an die Algorithmen zur Berechnung von authentisierten Hashes bei TLS.

⁴ In den Common Criteria steht »SFR« für *Security Functional Requirement* (Sicherheitsanforderung).

Generierte Tabellen

Im Fall des hier betrachteten eHealth-Konnektors gibt das BSI etwa 130 formale Sicherheitsanforderungen vor. [8] Wir zerlegen unser Produkt in 160 Module, aufgeteilt auf 24 Subsysteme und erreichbar über 65 Softwareschnittstellen. Auch ohne tiefes Verständnis der CC wird deutlich, dass die Relationen sehr komplex sind. Diese Relationen müssen in den Dokumenten in unterschiedlicher Form, beispielsweise in Tabellen, wiedergegeben werden.

Der Evaluierungsgegenstand wird parallel zum Schreiben der Dokumentation weiterentwickelt, so können sich während des Dokumentationsprozesses Anpassungen an der Modellierung ergeben. Diese Änderungen müssen in die Dokumente eingepflegt werden. Wenn sich beispielsweise herausstellt, dass eine Sicherheitsanforderung doch besser von einem anderen Modul erfüllt wird, muss diese Änderung gegebenenfalls an unterschiedlichen Stellen in mehreren Dokumenten nachgezogen werden. Das Risiko, dass Inkonsistenzen entstehen, ist sehr hoch. Manuell ist dies nur mit viel Sorgfalt und hohem Aufwand zu bewerkstelligen. Bei unserem Ansatz, die Relationen in einer Datenbank zu modellieren, ist die Änderung nur an einer Stelle – in der Datenbank – erforderlich. Danach werden die Dokumente neu erzeugt und der konsistente Stand ist wieder hergestellt.

Einbindung der Datenbank

Wir verwenden SQLite, ein flexibles und leichtgewichtiges Datenbanksystem. [9] Die Präambel der Dokumente enthält eine Umgebung `luacode`, in der die Datenbank initialisiert wird. Der Lua-Interpreter in Lua_T_E_X lädt dazu die `sqlite3` Bibliothek, die Funktionen zum Zugriff auf eine SQLite DB bereit stellt. [6] Die Datenbank wird im Speicher gestartet und die Tabellen angelegt. Dann werden die Tabellen mit den Daten bestückt. Die Daten liegen als kommaseparierte Dateien vor, im selben Git-Repository wie die Dokumente selbst. Der Zeitaufwand für das Starten und Provisionieren der Datenbank ist zu vernachlässigen, da sich alles im Hauptspeicher abspielt. Danach steht die Datenbank für Abfragen bereit. Wichtig ist, dass die Makros und Lua-Funktionen ausschließlich lesend auf die DB zugreifen: Nach dem Ende des Lua_T_E_X-Durchlaufs wird die Datenbank verworfen und beim nächsten Mal neu initialisiert.

Die Pflege der Datenbank erfolgt über direkte Anpassungen an den CSV-Dateien. Diese haben gegenüber einer (binären) Datenbankdatei den Vorteil, dass sie sich nachvollziehbar in Git versionieren lassen. Hier gab es die Abwägung, ob die Daten besser binär abgespeichert und dann mit einem Editor wie dem SQLite Browser bearbeitet werden, oder ob die Variante in »plain text« besser geeignet

ist, in der die CSV-Dateien editiert werden. Wir haben uns aufgrund der besseren Nachvollziehbarkeit der Änderungen für die Ablage als CSV-Dateien entschieden.

Zugriff auf die Datenbank

Der Zugriff auf die Datenbank erfolgt über eine Kette von Makros und Lua-Programmen. Dabei kommen mehrere Abstraktionen bzw. Delegationen zum Tragen. Ziel dieser Architektur ist es, den Lua-Anteil im L^AT_EX-Code möglichst gering zu halten. Das ist kein Selbstzweck: Die in Lua implementierten Funktionen der Plattform haben eine Komplexität erreicht, die – im Sinne einer sorgfältigen Softwareentwicklung – durch Unit-Tests abgesichert werden sollen.⁵ Dies ist jedoch mühselig, wenn der Lua-Code in luacode-Umgebungen oder `\directlua`-Makros eingebettet ist. Die Turnaround-Zeiten sind zu lang: Bis LuaT_EX gestartet und alle Pakete geladen sind, vergeht eine verhältnismäßig lange Zeit. Ein isolierter Lua-Interpreter (z. B. `texlua`) braucht nur Millisekunden für dieselben Tests. Außerdem gibt es Konflikte bei den Entwicklungswerkzeugen: Die Syntax-Unterstützung für Lua gerät aus den Fugen, wenn Lua-Code in einer `.tex`-Datei steht.

Im hier beschriebenen Verfahren dient eine *Brücke* als Adapter zwischen den L^AT_EX-Makros und dem Lua-Code. Die Brücke hat zwei Seiten: eine, die in einer luacode Umgebung in L^AT_EX formuliert ist und eine, die in einem Lua-Modul implementiert ist.



Als Beispiel soll hier das Makro `\sfrtext` dienen, das den Schlüssel eines DB-Eintrags erhält und für diesen Schlüssel den beschreibenden Text aus der DB liest und im Dokument ausgibt. Die folgenden Auszüge zeigen die Kette der Aufrufe.

Am Beginn der Kette steht die Definition des Makros `\sfrtext`, das vom Benutzer aufgerufen wird. Das Makro ist in der Datei `common/cc-macros.tex` definiert, die von allen Dokumenten eingelesen wird. Das Makro ruft die Lua-Funktion `getSfrText()` auf und übergibt den Parameter des Makros:

```
\newcommand{\sfrtext}[1]{\directlua{getSfrText("#1")}}
```

Die Lua-Funktion `getSfrText()` wird in der Datei `common/bridge_cc_core.tex` definiert, auf der L^AT_EX-Seite der Brücke. Hier wird an die gleich benannte Funktion im Lua-Modul `bridge_cc_core` delegiert, auf die Lua-Seite. Der Aufruf wird allerdings nicht 1:1 weitergegeben, sondern um die Datenstruktur `tex` ergänzt, die die Laufzeitumgebung des Textsatzprozesses enthält:

⁵ Das Lua-Paket `luaunit` bietet Funktionen dafür. [3]

```
\begin{luacode}

  bridge_cc_core = require "bridge_cc_core"

  function getSfrText(key)
    bridge_cc_core.getSfrText(key, tex)
  end

\end{luacode}
```

Die so aufgerufene Funktion in `lua/bridge_cc_core.lua` befindet sich auf der Lua-Seite der Brücke. Sie delegiert an eine Funktion gleichen Namens (diesmal im Modul `cc_core`). Das Ergebnis des Aufrufs wird mit `tex.sprint()` in den Ausgabestrom des LuaT_EX-Prozesses (und damit in das Dokument) geschrieben:

```
cc_core = require "cc_core"

function bridge_cc_core.getSfrText(key, tex)
  tex.sprint(cc_core.getSfrText(key))
end
```

In `lua/cc_core.lua` schließlich ist das SQL-Statement definiert. Die Datenstruktur enthält neben der Query auch noch Informationen über das Mapping des Resultats. Das Modul `cc_core` enthält das gesamte fachliche Wissen über ein Common Criteria Verfahren, die Modellierung der Entitäten in DB-Tabellen und die Zugriffsllogik dafür. Von hier aus wird schließlich die Datenbankabfrage gestartet:⁶

```
query={name="sfrtext", st=[[SELECT description FROM sfr WHERE label=? COLLATE
↪NOCASE]], resultitem = "description"}

function cc_core.getSfrText(key)
  return cmn.get_by_query_key("sfrtext", key)
end
```

Es ist zu beachten, dass die Funktionalität zum Auslesen der Datenbank komplett von LuaT_EX losgelöst ist und eigenständig aufgerufen werden kann. An dieser Stelle setzen Unit- und Integrationstests an, die außerhalb von LuaT_EX laufen.⁷

⁶ Tatsächlich kommen noch zwei weitere Lua-Module zum Tragen: `common` enthält Funktionen zum Vor- und Nachbearbeiten des Datenbankzugriffs, wie das Result Mapping. Das Modul `db_core` repräsentiert die Datenbank selbst, hat aber selbst keinen fachlichen Bezug zu Common Criteria. `db_core` bietet Funktionen zum Starten und Befüllen der Datenbank.

⁷ Um die Tests vollständig unabhängig von den Daten des Evaluierungsgegenstandes zu halten, gibt es eine Testdatenbank, die strukturell mit der Produktionsdatenbank identisch ist; deren Werte sich aber nicht ändern, wenn die Modellierung angepasst wird.

Continuous Integration / Continuous Delivery

Eine hervorragende Errungenschaft der agilen Softwareentwicklung ist die fortlaufende Integration von Arbeitsergebnissen in die Codebasis und die sofortige Auslieferung der Zwischenergebnisse. Für n-doc haben wir ein Verfahren entwickelt, das die Ideen der Continuous Integration (CI) und der Continuous Delivery (CD) auch für L^AT_EX-Dokumente umsetzt. Die Open-Source-Variante von n-doc nutzt GitHub-Actions. In der privaten Implementierung im hausinternen Projekt setzen wir auf Jenkins, einen populären CI-Server, und GitLab, ein System zur Verwaltung von Daten in Git.

Alle Anpassungen an den Dokumenten werden in separaten Branches in GitLab vorgenommen. So wird sicher gestellt, dass es geringe Zugriffskonflikte zwischen unterschiedlichen Redakteuren gibt. Sobald eine Anpassung (ein Feature) fertig entwickelt ist und mit einem »Merge Request« auf den Hauptstrang integriert wurde, löst das Versionierungssystem automatisch einen Build aller Dokumente in Jenkins aus – exakt so, wie auch die Software entwickelt und integriert wird.

Jenkins nutzt eine hierarchische Anordnung von Makefiles, um die Dokumente zu übersetzen. Auf der Ebene eines einzelnen Dokuments kommt wiederum latexmk zum Einsatz. Diese Übersetzungsvorgänge lassen sich sehr gut in mehreren Prozessen nebeneinander ausführen, obwohl T_EX an sich keine Parallelisierung unterstützt. Da jeder LuaT_EX-Prozess seine eigene Instanz der SQLite-Datenbank startet, gibt es keine Interferenzen durch konkurrierende Zugriffe.

Die Jenkins-Pipeline kopiert die PDF-Dateien auf einen Server im Intranet, auf dem eine spezielle Website liegt. Die Website wird ebenfalls neu generiert, um den aktuellen Stand der Dokumente wiederzugeben. Dort ist nicht nur der aktuelle Arbeitsstand (Snapshot) abrufbar, sondern auch alle bislang an den CC-Evaluator ausgelieferten Versionen. So lässt sich lückenlos nachverfolgen, wann welche Arbeitsergebnisse übergeben wurden. Die übliche Frage »Wo liegt denn die aktuelle Version?« (bekannt aus vielen Softwareprojekten) stellt sich damit nicht mehr. Die aktuelle Version wird automatisch erzeugt und liegt an einer festen Stelle.

Versionierung

Als Softwareentwickler wissen wir um die Kritikalität eines klar definierten Versionierungssystems für den Erfolg eines Projekts. Mit »Versionierungssystem« ist jedoch nicht nur die Verwendung von Git (o. ä.) gemeint, sondern auch ein Schema für die Vergabe von Versionsnummern.

Dabei orientieren wir uns an dem, was im Java-Umfeld seit Jahren durch das Build-Management-Werkzeug Maven etabliert ist: Zwischenstände enthalten in ihrer Ver-

sionsnummer das Suffix »-SNAPSHOT«. Ein Dokument der Version »1.6-SNAPSHOT« ist zu lesen als: »Ein Zwischenstand vor der Version 1.6«. Wenn ein Release erzeugt wird, entfernt man das Suffix »-SNAPSHOT«. Im Versionsverwaltungssystem wird ein Tag »v1.6« erzeugt. Damit steht der neue Versionsstand fest und ist reproduzierbar. Schließlich wird die Versionsnummer inkrementiert (»1.7«) und das Suffix wieder angehängt (»1.7-SNAPSHOT«). Damit befindet man sich »auf dem Weg zur Version 1.7«. Was hier gewöhnungsbedürftig klingt, ist für Entwicklerinnen und Entwickler Teil der täglichen Arbeit und deckt sich zu 100% mit dem, was auch in der Softwareentwicklung praktiziert wird.

Die Versionen der Dokumente werden in einer Tabelle der Datenbank (also einer CSV-Datei) gepflegt. Der Releaseprozess zur Manipulation dieser Tabelle ist als Shell-Script implementiert, sodass bei diesem kritischen Vorgehen Flüchtigkeitsfehler weitgehend ausgeschlossen sind.

Paketierung der Werkzeuge als Docker-Image

n-doc besteht aus zwei Teilen: Erstens einem Satz von Beispieldokumenten, die als Vorlage für eigene Dokumentationsprojekte verwendet werden können. Zweitens die verwendeten Werkzeuge: LuaT_EX, die benötigten Makro-Pakete, SQLite und weitere Werkzeuge. Alle diese Open Source-Werkzeuge haben wir in einem Docker-Image zusammengestellt. Docker bietet sich als Paketierungsverfahren an. Obwohl n-doc aus einer Menge Tools besteht, werden sie in einer vom Benutzer leicht handhabbaren Einheit verteilt, sodass sie ohne großen Installationsaufwand verwendet werden können. Aus Sicht des Administrators können leicht neue Versionen der Images verteilt werden (z. B. wenn ein neues L^AT_EX-Makropaket hinzugefügt werden soll), die über private oder öffentliche Docker-Repositorys bereitgestellt werden.

Wir haben uns für T_EX Live als Distribution entschieden. Problematisch daran ist lediglich die Größe: Eine volle Installation plus die ansonsten benötigten Programme nimmt leicht ein paar Gigabyte in Anspruch. Gerade wenn die Docker-Images über das Internet verteilt werden, kann dies mit langen Ladezeiten einher gehen.

Stattdessen haben wir eine T_EX Live Installation zusammengestellt, die ausschließlich die benötigten Pakete enthält. n-doc benötigt ca. 150 Pakete, die direkt oder als indirekte Abhängigkeit geladen werden. Die Installation von T_EX Live ist minimal: Startpunkt ist das *infra*-Schema, das lediglich die Verzeichnisstruktur bereitstellt. Die Flexibilität des Installationstools *tlmgr* unterstützt dieses Vorgehen, sodass das komplette Image weniger als 600 MB umfasst – inklusive Betriebssystem und aller benötigten Utilities wie SQLite, Git, Perl oder Python-Pygments (für das Paket *minted* zum Ausdruck von Listings).

Veröffentlichung der Plattform

n-doc ist als hausinternes Projekt entwickelt worden; zuerst als kleines Experiment und dann mit voller Unterstützung durch die Leitungsebenen unseres Unternehmens. Die Resonanz bei den Autorinnen und Autoren war sehr positiv. Der beschleunigte Fortschritt bei der Erstellung der Dokumentation und die qualitativ besseren Ergebnisse wurden deutlich sichtbar.

Eher durch Zufall ergab sich die Gelegenheit, das System auf einer Konferenz zur IT-Sicherheit zu präsentieren. [5]. Dort wurde deutlich, dass auch andere Unternehmen Schwierigkeiten mit Office-basierten Zertifizierungsverfahren haben und Bedarf an innovativen Lösungen im Umfeld der CC-Dokumentation besteht. Im Anschluss an diese Konferenz entstand die Idee, die Plattform unter einer Open-Source-Lizenz zu veröffentlichen. Dazu waren zwei Schritte notwendig: Die Auswahl der geeigneten Lizenz und – wichtiger noch – die Bereitstellung in einer angemessenen Form.

Bei der Lizenz haben wir uns für die MIT-Lizenz entschieden. Diese sehr liberale Lizenz hat für die nutzenden Unternehmen den Vorteil, dass sie ihre Anpassungen an n-doc nicht veröffentlichen müssen. Das hätte vermutlich sehr negative Auswirkungen auf die Akzeptanz der Plattform: Produkte, die nach Common Criteria zertifiziert werden, enthalten in der Regel proprietären Code und vertrauliche Sicherheitseigenschaften. Wenn die Dokumentation dieser Eigenschaften veröffentlicht werden müsste, würde sich vermutlich niemand für eine solche Dokumentationsplattform entscheiden. Die MIT-Lizenz erlaubt explizit die Entwicklung von Derivaten ohne Zwang zur Veröffentlichung.

Für die Bereitstellung in angemessener Form hat es sich bewährt, einen Satz Beispieldokumente zu entwickeln, die einen fiktiven Evaluierungsgegenstand beschreiben, einen VPN-Clients. Dieses Produkt ist natürlich nicht implementiert, doch die Dokumente bilden einen konsistenten Satz an Herstelldokumenten ab. Dies ermöglicht Nutzern von n-doc, die Zusammenhänge zwischen Dokumenten und Datenbanken zu erlernen und die Dokumentvorlagen Stück für Stück an die eigenen Bedürfnisse anzupassen und den Evaluierungsgegenstand zu beschreiben.

Die Beispielanwendung nennen wir »n-doc template«. Sie ist bei GitHub veröffentlicht. [4] In einem benachbarten Repository befinden sich die Quellen für das Docker-Image. Das Image selbst ist unter dem Tag `ndesign/n-doc` bei Docker Hub veröffentlicht (<https://hub.docker.com/r/ndesign/n-doc>).



Fazit und Ausblick

In der Ausgabe 2/2020 der *T_EXnischen Komödie* geht Lukas C. Bossert auf »Documentation as Code« im Kontext des Literate Programming ein. [1] Wir fügen diesem

Konzept eine weitere Konnotation hinzu und begreifen »Documentation as Code« so, dass die Dokumentation nicht Teil des Codes ist, sondern selbst wie Code behandelt wird. Die »Best Practices« des Software-Engineering werden auf die Erstellung von Dokumentationen übertragen: Separation of Concerns (hier werden Layout, Navigation und Inhalt durch getrennte Funktionen behandelt), Versionierung und Continuous Integration/Delivery sind Prinzipien, die sich in den letzten Jahren als Vorgehensmodelle und Werkzeuge in der Softwareentwicklung bewährt haben. Wir greifen diese Prinzipien auf und wenden sie auf die Erstellung der Dokumentation an. So entstehen Arbeitsabläufe, die Entwicklerinnen und Entwicklern vertraut sind. Die hohe Qualität der Ergebnisse wird durch L^AT_EX als technische Basis der Dokumentationsplattform sichergestellt.

Mit dem n-doc-Verfahren haben wir einen Werkzeugkasten entwickelt und veröffentlicht, mit dem Unternehmen ihre technische Dokumentation in Zertifizierungsverfahren bewältigen können.

n-doc lässt sich nicht nur für Zertifizierungen nach Common Criteria nutzen: Wir haben ebenfalls Zertifizierungsverfahren nach technischen Richtlinien zu absolvieren. Für diese Verfahren haben wir die spezifischen Makros, die Brücke und das cc_core Modul durch entsprechende Funktionalität für technische Richtlinien ersetzt. Die Anwendung auf andere Zertifizierungsschemata wie IT-Grundschutz ist vorstellbar.

Literatur und Software

- [1] Lukas C. Bossert: »Verschiedene Möglichkeiten zum Kommentieren und Dokumentieren von Code«, *Die T_EXnische Komödie*, 32. Jahrgang (2 2020), 36–48.
- [2] Eric J. Evans: Domain-Driven Design, Tackling Complexity in the Heart of Software, Addison-Wesley, 2003.
- [3] Philippe Fremy: LuaUnit, <https://github.com/bluebird75/luauunit>.
- [4] n-design GmbH, Hrsg.: n-doc, Vorlagen für die Herstellerdokumente in einem fiktiven CC-Verfahren der Stufe EAL3+, 2020, https://github.com/n-design/n-doc_template.
- [5] Alexander Krumeich in Tagungsband 29. SmartCard Workshop, (Hrsg.: Ulrich Waldmann), Fraunhofer SIT, Darmstadt, 2019.
- [6] LuaSQLite 3, a Lua 5.1 to 5.3 wrapper for the SQLite3 library, <http://lua.sqlite.org/index.cgi/home>.

- [7] The Common Criteria Recognition Agreement Members: Common Criteria for Information Technology Security Evaluation, Part 1: Introduction and general model, Common Criteria, Version 3.1R5, Common Criteria Portal, 2017, <http://www.commoncriteriaportal.org/thecc.html>.
- [8] Bundesamt für Sicherheit in der Informationstechnik: Schutzprofil 2: Anforderungen an den Konnektor, BSI-CC-PP-0098, Common Criteria Schutzprofil (Protection Profile), Version 1.5.4, Bundesamt für Sicherheit in der Informationstechnik (BSI), 2020.
- [9] SQLite, 2020, <https://www.sqlite.org>.

Professionell präsentieren mit der Beamer-Klasse

Pascal Braband

L^AT_EX in Kombination mit der Beamer-Klasse liefert ein mächtiges Werkzeug, um beeindruckend professionelle (wissenschaftliche) Präsentationen zu erstellen. Doch wenn es an das eigentliche Vorführen dieser Präsentationen geht, sind die Möglichkeiten oft limitiert. So steht standardmäßig erstmal lediglich ein simpler PDF-Viewer zur Verfügung, mit dem die Folien dargestellt werden.

Für einfache Präsentationen mag dies genügen, jedoch können bei komplexeren, längeren Präsentationen mehr Funktionen nützlich sein. Genau für diese Ansprüche gibt es einige wenige Programme, die sich darauf spezialisiert haben, PDF-Dateien der Beamer-Klasse professionell zu präsentieren. Drei solcher Lösungen werden in diesem Artikel vorgestellt.

Herausforderungen beim Präsentieren

Ein normaler PDF-Viewer ist in der Hinsicht limitiert, dass die einzige Möglichkeit, Dokumente zu präsentieren, darin besteht, diese im Vollbildmodus auf dem eigenen Rechner zu öffnen und den Bildschirm dann auf das Präsentationsgerät (z. B. ein Beamer) zu spiegeln. So ist der eigene Bildschirm quasi komplett belegt, da der Zuhörer nun alles sieht, was der Vortragende auf seinem Rechner macht. Zusätzliche Fenster mit Notizen können so nicht geöffnet werden und auch die Such- und Übersichtsfunktionen des PDF-Viewers sind nicht mehr unbemerkt verwendbar.

Eine weitere Schwierigkeit, die sich besonders bei mit Beamer erstellten PDF-Dateien ergibt, ist das Darstellen der Notizen. Nicht jeder PDF-Viewer kann mit dem Notizenformat von Beamer umgehen, bei welchem eine PDF-Seite sowohl die Präsentationsfolie als auch die Notizenfolie enthält.

Wünschenswert ist also ein Programm, welches mit Beamer PDF-Dateien mit Notizen korrekt darstellen kann und zusätzlich auf dem eigenen Bildschirm mehr Informationen zum Präsentieren anzeigt als auf dem Präsentationsgerät. Im Grunde genommen wäre dies das Gleiche, was PowerPoint und Keynote für ihre eigenen Formate machen, nur für PDF-Dokumente.

Vorstellung von SlidePilot

Genau zu diesem Zweck wurde SlidePilot für macOS entwickelt. Das Programm fasst zusätzliche Werkzeuge und Informationen, die beim Präsentieren nützlich sind, in einem Moderatorenfenster zusammen. Dieses wird auf dem eigenen Bildschirm angezeigt, wobei die Präsentation selbst auf einem angeschlossenen, zweiten Bildschirm dargestellt wird.

SlidePilot richtet sich im Wesentlichen an Benutzer, die Präsentationen, erstellt mit der Beamer-Klasse, vorführen möchten. So bietet das Programm vor allem die Möglichkeit, eine PDF-Datei so zu spalten, dass Präsentations- und Notizenfolie einer PDF-Seite korrekt getrennt und separat dargestellt werden. Im Moderatorenfenster kann dann unter anderem die Notizenfolie dargestellt werden, während im Präsentationsfenster das dazugehörige Gegenstück gezeigt wird. Mit einem entsprechenden PDF-Keyword erkennt SlidePilot automatisch, wie das Dokument gespalten werden muss. Das folgende simple Beispiel zeigt die Darstellung von Notizen mit Beamer inklusive dem speziellen PDF-Keyword:

```
\documentclass{beamer}
\usepackage{pgfpages}
\setbeameroption{show notes on second screen=right}

% Notizenposition Hinweis (andere Positionen analog)
\pdfinfo{/Keywords (SP-Right)}

\begin{document}
\frame{
  Inhalt ...
}
\note{
  Notizen ...
}
\end{document}
```

Es ist genauso gut möglich, beliebige andere PDF-Dateien zu präsentieren. Für diesen Fall bietet SlidePilot die zusätzliche Möglichkeit an, Notizen im `rtf`-Format mit dem eingebauten Notizen-Editor hinzuzufügen, zu bearbeiten und zu formatieren.

SlidePilot bietet für das Moderatorenfenster verschiedene Layouts. Neben den jeweiligen Notizen kann außerdem die aktuelle Folie und die nächste Folie angezeigt werden. Weiterhin wird die aktuelle Uhrzeit gut sichtbar dargestellt und die Vortragszeit kann über eine Stoppuhr oder einen Timer verfolgt werden. Eine Übersicht über alle Folien der Präsentation kann ebenfalls über eine Seitenleiste eingeblendet werden.

Zum Präsentieren bietet das Programm weitere nützliche Werkzeuge. Dazu zählt ein (Laser-) Pointer, das Verhüllen der Präsentation mit einem weißen oder schwarzen Vorhang und das Zeichnen auf der aktuellen Folie. Weitere Informationen zum Programm und seiner Benutzung lassen sich in der ausführlichen, englischsprachigen Dokumentation finden (<https://slidepilot.gitbook.io>). SlidePilot ist auf der offiziellen Website (<https://slidepilotapp.com>) kostenlos zum Download verfügbar. Allerdings ist es möglich, über die Website die Entwicklung mit Spenden zu unterstützen.



Abb. 1: Moderatorenfenster in SlidePilot

Alternativen

SlidePilot wurde primär für macOS entwickelt, eine Ausweitung auf Linux bzw. Windows ist momentan nicht geplant. Es gibt allerdings gute Alternativen für die anderen Betriebssysteme. Recht weit verbreitet ist das Programm pdfpc (<https://github.com/pdfpc/pdfpc>), welches ähnliche Funktionen wie SlidePilot bietet. Am einfachsten funktioniert die Installation unter Linux, aber auch unter Windows ist es möglich, pdfpc zu installieren. Eine andere Alternative ist pympress (<https://github.com/Cimbali/pympress>). Dieses Programm wurde in der Programmiersprache Python entwickelt und kann daher sehr einfach über Pythons Paketverwaltung installiert werden.



Chaotische Symmetrien mit Lua berechnet

Herbert Voß

Chaotische Symmetrien lassen sich grafisch besonders gut darstellen. Mit der Anwendung von $\text{\TeX}$ Lua als Programm, erfordert es noch nicht einmal die Installation der Skriptsprache Lua; diese ist standardmäßig bei einer Vollinstallation von $\text{\TeX}$ automatisch vorhanden und kann sowohl aus einem $\text{\LaTeX}$ -Dokument heraus als auch extern mit $\text{\TeX}$ Lua benutzt werden.

Einführung

Anfang der neunziger Jahre des letzten Jahrhunderts gab es eine geradezu euphorische Behandlung chaotischer Systeme. [5] So bemerkte ich 1994:

Chaos und Fraktale sind seit einiger Zeit aus dem Computeralltag nicht mehr wegzudenken. Wenn schon das im Literaturbereich anzusehnde Kursbuch des Berliner Rotbuch-Verlags dem Chaos ein ganzes Heft von 162 Seiten widmet, im Nachrichtenmagazin »Der Spiegel« eine Serie über drei Hefte (!) hinweg mit dem Titel »Mythos im Computer« erscheint [1, 2, 3], und der Fernseh-Kulturkanal Arte diesem Thema einen ganzen Abend mit verschiedenen Beiträgen widmet, dann gehört die Theorie vom Chaos und den Fraktalen mit Sicherheit in den Kultbereich. [8]

Durch Zufall fielen mir meine alten Unterlagen zu diesem Thema in die Hände, sodass ich die Beispiele, die sämtlichst mit Turbo-Pascal erstellt wurden, auf Lua übertragen habe. Dabei versuche ich stets, die Berechnungen auch aus einem Dokument heraus durchführen zu können. Die große Aufmerksamkeit für das Thema

wurde natürlich durch das sogenannte »Apfelmännchen«, welches Benoit Mandelbrot bei seinen Forschungen zufällig entdeckte, hervorgerufen. [6, 7] Die einfache quadratische Gleichung

$$z_{n+1} = z_n^2 + C, \quad z, c \in \mathbb{C}$$

startet beispielsweise mit $C = 0,1 + i0,2$ und $z_0 = 0 + i0$ und wird dann solange wiederholt (rekursiv) angewendet, bis der Betrag des Ortsvektors von $\vec{z}$ einen vorgegebenen Radius verlässt. Die Anzahl der dafür notwendigen Schritte bestimmt dann die Farbe des Startpunktes in der komplexen Ebene. Wird nach einer vorgegebenen Zahl an Durchläufen der Radius nicht verlassen, erfolgt ein Abbruch und der Punkt bekommt dann immer dieselbe Farbe. Für ein Koordinatensystem in der komplexen Ebene geht man jetzt einfach alle Werte durch und ersetzt jeden Punkt mit der Farbe des Ergebnisses.

Mandelbrotmenge

Die Berechnungen für die Mandelbrotmenge kann man innerhalb des Dokuments ausführen und der Einfachheit halber eine Grafik im Format .ppm erstellen, deren Struktur ziemlich trivial ist. Damit sie gleich ins Dokument mit `\includegraphics` eingefügt werden kann, wird sie am Ende mit dem Programm `pnmtopng` in das Grafikformat .png umgewandelt. Dazu muss das Dokument mit der Option `--shell-escape` übersetzt werden; sonst wäre der Aufruf externer Programme nicht möglich. Die Lua-Bibliothek `complex` ist grundsätzlich nicht nötig, gestaltet aber den Algorithmus etwas übersichtlicher. Sie kann unter <http://lua-users.org/wiki/ComplexNumbers> heruntergeladen und im lokalen $\text{\TeX}$ -Baum gespeichert werden. Für die Zeichen `%` und `\` muss im Luacode als Präfix `\string` benutzt werden.



```
\begin{luacode}
local complex = require ("complex")
local size = 500
local im = {}
for r = 0, size-1 do
  im[r] = {}
  for c = 0, size-1 do im[r][c] = 0 end
end

local palette = {}
for i = 0, 255 do
  palette[i] = tostring(255-i).." "..tostring(255-i).." "..tostring(255-i).." "
end

local xMin = -2
```

```

local xMax = 0.5
local yMin = -1.25
local yMax = 1.25
local dx = (xMax-xMin)/size
local dy = (yMax-yMin)/size
local Radius = 5

local function getRelX(i) return i*dx+xMin end
local function getRelY(i) return i*dy+yMin end

for col = 0, size-1 do
  for row = 0, size-1 do
    c = complex.new(getRelX(col), getRelY(row)) -- Startpunkt
    z = complex.zero -- Startwert
    local Color = 0 -- Trefferzähler
    while (z:abs() < Radius) and (Color < 511) do
      z = z^2 + c
      Color = Color+1
    end
    im[row][col] = Color \string% 256 -- modulo 256
  end
end

outFile = io.open("apfel.ppm", "w+") -- Dateiname
outFile:write("P3 ") -- Grafiktyp R G B
outFile:write(size.." ..size.." ") -- Anzahl Pixel: Breite x Höhe
outFile:write("255\string\n") -- Höchster Zahlenwert (0..255)
for row = 0, size-1 do -- alle Zeilen
  for col = 0, size-1 do -- alle Spalten
    local colorIndex = im[row][col] -- Trefferzahl
    outFile:write(palette[colorIndex].." ") -- Farbe(Trefferzahl)
  end
  outFile:write("\string\n") -- neue Zeile
end
outFile:close() -- Datei schließen
os.execute("pnmtopng apfel.ppm > apfel.png") -- ppm -> png Format
\end{luacode}

\begin{figure}\centering
\includegraphics[width=0.5\linewidth]{apfel}
\caption{Ausgabe des Luaprogramms}
\end{figure}

```

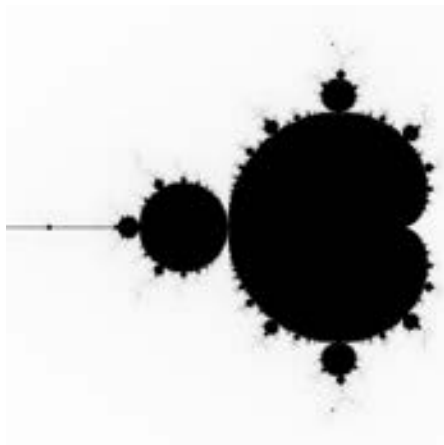


Abb. 1: Ausgabe des Luaprogramms

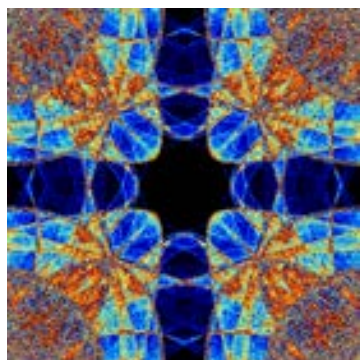
Chaotische Symmetrien

Als Folge der damaligen Euphorie wurden zahlreiche Bücher zu den Themen Chaos, Attraktoren, Symmetrien in der komplexen Ebene usw. veröffentlicht, unter anderem auch [4]. Der mathematische Aspekt rückte immer mehr in den Hintergrund und die Grafikausgabe in den Vordergrund. Man musste nichts weiter tun, als eine Gleichung und entsprechende Parameter zu finden. Die sogenannten Quiltmuster werden dabei durch folgende Gleichung mit $z = x + iy$ erzeugt:

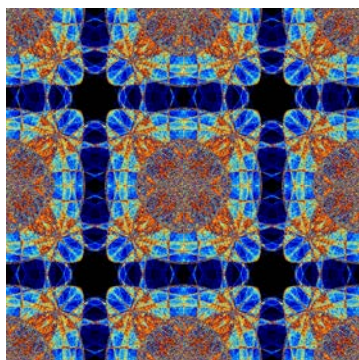
$$\begin{aligned}
 f(z) = & m \cdot z + \nu + \lambda(\sin 2\pi x + i \sin 2\pi y) \\
 & + \alpha(\sin 2\pi x \cos 2\pi y + i \sin 2\pi y \cos 2\pi x) \\
 & + \beta(\sin 4\pi x + i \sin 4\pi y) \\
 & + \gamma(\sin 6\pi x \cos 4\pi y + i \sin 6\pi y \cos 4\pi x)
 \end{aligned} \tag{1}$$

Hierbei sind $\lambda, \alpha, \beta, \gamma$ reelle Zahlen und ν als komplexe Zahl entweder Null oder $0,5 + i0,5$. Die Zahl der Iterationen wird jetzt vorgegeben und man ordnet jedem Punkt der komplexen Ebene die Zahl der »Treffer« zu, die wiederum die Farbe des Punktes bestimmt. Die Abbildung 2a zeigt eine Ausgabe für zehn Millionen Iterationen mit den dort angegebenen Parametern.

Die einzelnen Abbildungen können bei symmetrischen Quiltmustern einfach durch Wiederholungen zu einem großflächigen Muster zusammengesetzt werden, da sie achsensymmetrisch sind. Dies gilt nicht mehr für die im Folgenden beschriebenen hexagonalen Muster.

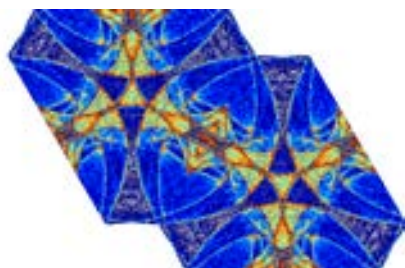


(a) Erzeugung des Musters

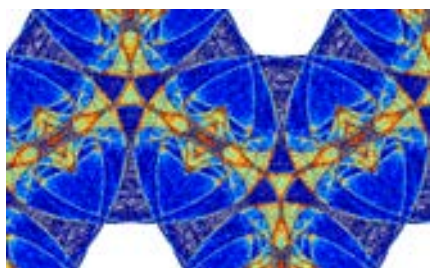


(b) Ausgabe als Kachelmuster

Abb. 2: Ausgabe eines symmetrischen Quilmusters mit folgenden Parametern:

 $\lambda = -0.59$, $\alpha = 0.2$, $\beta = 0.1$, $\gamma = -0.33$, $\omega = 0$, $m = 2$ und $\nu = 0 + i0$.


(a) Erzeugung eines hexagonalen Quilts



(b) Ergänzung zu einem Rechteck

Abb. 3: Ausgabe eines hexagonalen Quilmusters nach [4].

Die Gleichung 1 wird noch komplizierter, wenn man das Muster auf ein hexagonales Gitter bezieht, sodass die Ausgabe in einem schiefwinkligen Koordinatensystem erfolgt, bei dem die Achsen einen Winkel von 60° einschließen. Nach fünf Millionen Iterationen ergibt sich dann beispielsweise Abbildung 3, welche man nachträglich zu einem Rechteck ergänzen kann. Die Farben ergeben sich durch die gewählte Palette, die maßgeblich für das ästhetische Moment der Abbildungen ist.

Dieses Rechteck lässt sich jetzt einfach durch Spiegelung an den Achsen zu einem größeren Muster zusammensetzen, sodass man die unterschiedlichsten Ausgaben erreichen kann (Abb. 4). Das Lua-Programm wurde dieses Mal extern mit `texlua --shell-escape <file>` ausgeführt, wobei ebenfalls nur die externe Biblio-

thek `complex.lua` benutzt wurde und die Grafiken wieder über das Format `.ppm` erzeugt wurden. Durch die Verwendung von $\text{\TeX}$ Lua muss man sich nicht um die Pfade kümmern, alle Lua-Bibliotheken, die in der $\text{\TeX}$ -Distribution vorhanden sind, können einfach genutzt werden, ebenso wie die in lokalen $\text{\TeX}$ -Verzeichnissen vorhandenen Lua-Dateien.

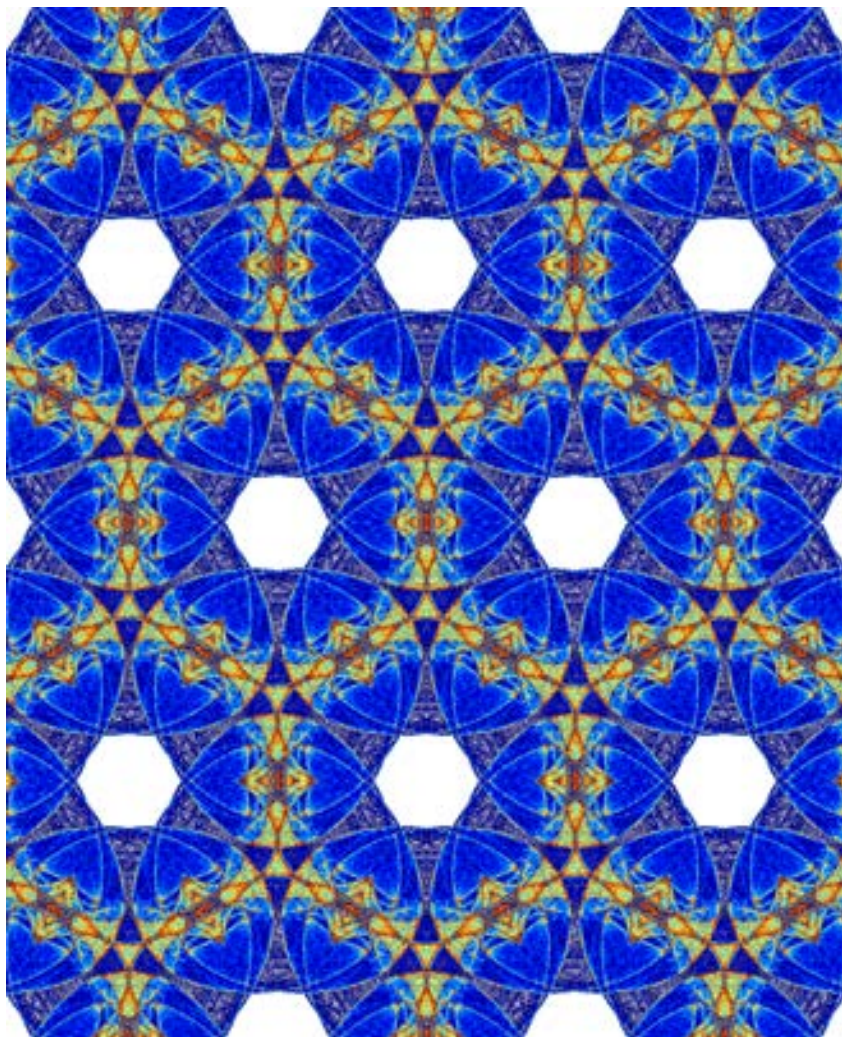


Abb. 4: Ausgabe als großflächiges Muster durch Spiegelungen des Rechtecks aus Abb. 3b.

Literatur und Software

- [1] Peter Brügge: »Mythos aus dem Computer – Teil I«, *Der Spiegel*, 39 (1993), 156–164.
- [2] — »Mythos aus dem Computer – Teil II«, *Der Spiegel*, 40 (1993), 232–241.
- [3] — »Mythos aus dem Computer – Teil III«, *Der Spiegel*, 41 (1993), 240–252.
- [4] Michael Field, Martin Golubitsky: Chaotische Symmetrien: Die Suche nach Mustern in Mathematik, Kunst und Natur, Birkhäuser, Basel, Boston und Berlin, 1993, S. 220, ISBN: 9783034862271, <https://books.google.de/books?id=ptKcBgAAQBAJ>.
- [5] Ordnung und Chaos in der unbelebten und belebten Natur, Verhandlungen der Gesellschaft Deutscher Naturforscher und Ärzte, 2. Aufl., (Hrsg.: Wolfgang Gerok u. a.), 115. Versammlung, 17. bis 20. Sept. 1988, Freiburg i. Br., S. Hirzel, Wissenschaftliche Verlagsgesellschaft, Stuttgart, 1990, S. 441, ISBN: 3-8047-1144-8, https://primo.fu-berlin.de/FUB:FUB_ALMA_DS21789295380002883.
- [6] Benoît B. Mandelbrot: Die fraktale Geometrie der Natur, Birkhäuser, Basel, Boston und Berlin, 1994, S. 491, ISBN: 9783034850278, <https://books.google.de/books?id=ayGdBgAAQBAJ>.
- [7] Heinz-Otto Peitgen, Hartmut Jürgens, Dietmar Saupe: Chaos: Bausteine der Ordnung, Klett-Cotta und Springer-Verlag, Stuttgart u. a., 1994, S. 688, ISBN: 9783608954357, <https://books.google.de/books?id=J5HjLQAACAAJ>.
- [8] Herbert Voß: Chaos und Fraktale selbst programmieren: von Mandelbrotmen- gen über Farbmanipulation zur perfekten Darstellung, Franzis-Verlag, Poing, 1994, S. 364, ISBN: 9783772370038, https://books.google.de/books?id=%5C_1NwHAAACAAJ.

Von fremden Bühnen

Neue Pakete auf CTAN

Jürgen Fenn

Der Beitrag stellt neue Pakete auf CTAN seit der letzten Ausgabe bis zum Redaktionsschluss in umgekehrter chronologischer Reihenfolge vor. Bloße Updates können auf der moderierten *CTAN-ann*-Mailingliste verfolgt werden.

ektype-tankā von *Niranjan Vikas Tambe* enthält vier hochwertige Fonts von EkType zum Setzen von Texten in der indischen Schrift Devanagari.

CTAN: fonts/ektype-tankā

namedef von *Phelype Oleinik* ermöglicht es, Parameter in T_EX-Makros mit ihrem Namen anzusprechen, also semantisch, anstelle ihrer Nummer in der Definition. Dadurch lassen sich Makros, vor allem längere Makros, viel leichter lesen.

CTAN: macros/generic/namedef

menucard von *Sascha Frank* stellt Makros zum Setzen einfacher Speisekarten bereit.

CTAN: macros/latex/contrib/menucard

mlmath von *Ma Zheng* schlägt eine Notation für Algorithmen zum maschinellen Lernen vor.

CTAN: macros/latex/contrib/mlmath

spix von *Louis Paternault* ist eine weitere L^AT_EX-Kompilierhilfe, die man aus dem T_EX-File heraus aufrufen kann.

CTAN: support/spix

shththesis von *Li Rundong* ist eine inoffizielle Vorlage für Abschlussarbeiten an der ShanghaiTech University.

CTAN: macros/latex/contrib/shththesis

hvarabic von *Herbert Voß* bietet eine Unterstützung für den Satz von Schriften, die von rechts nach links laufen, insbesondere Arabisch.

CTAN: hvarabic

commutative-diagrams von *Paolo Brasolin* enthält eine Bibliothek zum Zeichnen von kommutativen Diagrammen mit TikZ.

CTAN: graphics/pgf/contrib/commutative-diagrams

courierten von *Daniel Benjamin Miller* enthält die Schriftart Courier 10 Pitch BT von Bitstream in den Formaten OpenType und Type1 nebst der dazugehörigen L^AT_EX-Unterstützung.

CTAN: fonts/courierten

kpfonts-otf von *Daniel Flipo* enthält die OpenType-Fassung der Kp-Fonts von *Christophe Caignaert*. Sie können mit LuaT_EX und XeT_EX verwendet werden.

CTAN: fonts/kpfonts-otf

conditext von *Jean-Louis Brabant* ist ein weiteres Paket, mit dem man bestimmte Inhalte bedingt in ein L^AT_EX-Dokument einbringen kann.

CTAN: macros/latex/contrib/conditext

beamertheme-pure-minimalistic von *Kai Norman Clasen* verspricht ein »wirklich minimalistisches Beamer-Thema«.

CTAN: macros/latex/contrib/beamer-contrib/themes/beamertheme-pure-minimalistic

semantex von *Sebastian Ørsted* ist die Alpha-Version einer neuen objektorientierten Notation für den Mathematiksatz, damit man auch bei mathematischen Ausdrücken eine generische Auszeichnung vornehmen kann.

CTAN: macros/latex/contrib/semantex

biblatex-vancouver von *Agnibho Mondal* enthält den BibL^AT_EX-Stil Vancouver, der auf Numeric beruht.

CTAN: macros/latex/contrib/biblatex-contrib/biblatex-vancouver

tile-graphic von *Donald P. Story* zerteilt eine vorgegebene Grafikdatei in rechteckige Teile und fügt sie wieder in einer oder in mehreren PDF-Dateien zusammen.

CTAN: macros/latex/contrib/tile-graphic

utf8add von *Palle Jørgensen* erweitert die Auswahl an UTF8-Zeichen, die unmittelbar in L^AT_EX-Dateien eingegeben werden können, auch beim Mathematiksatz.

CTAN: macros/latex/contrib/utf8add

tikz-lake-fig von *Joseph Stachelek* stellt Makros bereit, mit denen man Seen, Verbindungen zwischen Wasserflächen, Wasserstände und dergleichen schematisch in Grafiken darstellen kann.

CTAN: graphics/pgf/contrib/tikz-lake-fig

biblatex-unified von *Kai von Fintel* ist ein BibL^AT_EX-Stil, der die Vorgaben des *Unified Style Sheet for Linguistics Journals* der *Linguistic Society of America* umsetzt.

CTAN: macros/latex/contrib/biblatex-contrib/biblatex-unified

pwebmac von *Andreas Scherer* bietet ein konsolidiertes Makro `pwebmac.tex` für alle $\text{\TeX}$ -Engines für das WEB-System.

CTAN:web/pwebmac

marathi von *Niranjan Vikas Tambe* bietet eine Grundausstattung zum Setzen von Texten in der indischen Sprache Marathi.

CTAN:language/marathi

bookshelf von *Peter Flynn* ist ein schönes Paket, das aus den Titeldaten in einer Bib $\text{\TeX}$ -Datei eine Grafik erzeugt, die ein Bücherregal mit den entsprechenden Bänden zeigt. Man kann sie als Hintergrund für Videokonferenzen mit Zoom einsetzen – eine Gelegenheit, bei der sich die Teilnehmer oft vor ihrem gut gefüllten Bücherregal zeigen.

CTAN:graphics/bookshelf

akshar von *Vũ Văn Dũng* verbessert den Umgang mit Silben beim Satz von Text in der indischen Schrift Devanagari.

CTAN:macros/latex/contrib/akshar

helmholtz-ellis-ji-notation von *Thomas Nicholson* setzt die *Helmholtz-Ellis Ji Pitch Notation* in der Fassung von 2020 (HEJI2) in $\text{\LaTeX}$ um. Das Paket, das $\text{\XeLaTeX}$ voraussetzt, ermöglicht es, Zeichen, die für den Notensatz in Pythagoreischer Stimmung benötigt werden, im Fließtext, in Beschriftungen etc. einzufügen.

CTAN:fonts/helmholtz-ellis-ji-notation

glossaries-nynorsk von *Eirik Haustveit* enthält die für das Paket `glossaries` von *Nicola Talbot* die Sprachunterstützung »Nynorsk«.

CTAN:macros/latex/contrib/glossaries-nynorsk

vcell von *Xavier Dionne* stellt Befehle bereit, mit denen man den Inhalt von Tabellenzellen vertikal ausrichten kann.

CTAN:macros/latex/contrib/vcell

readablecv von *Philip Stone* ist eine weitere Klasse zum Setzen von Lebensläufen und Begleitschreiben.

CTAN:macros/latex/contrib/readablecv

beamerappendixnote von *Christoph Semken* stellt einen Befehl und eine Umgebung bereit, mit der man in einer Beamer-Präsentation Anmerkungen in einen »Anhang« auslagern kann.

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

lua-uni-algos von *Marcel Krüger* hat das Ziel, Bibliotheken aufzubauen, die es leichter machen, Unicode-Zeichen mit Lua zu verarbeiten.

CTAN:macros/luatex/generic/lua-uni-algos

antanilipsum von *Jack Coleman* wandelt das Paket `kantlipsum` von *Enrico Gregorio* ab, welches Blindtexte nach *Kant* ausgibt, und erzeugt stattdessen unsinnige

Spaßsätze aus der italienischen Fernsehserie *Amici Miei*.

CTAN:macros/latex/contrib/antanilipsum

tikztosvg von *Pablo Emilio Escobar Gaviria* enthält ein Bash-Skript, mit dem man mittels $\text{X}\text{\TeX}$ und pdf2svg Grafiken, die mit TikZ erstellt worden sind, nach SVG wandeln kann.

CTAN:support/tikztosvg

anonymous-acm von *Brett A. Becker* vereinfacht es, anonymisierte Fassungen von Aufsätzen zu einer ACM-Zeitschrift für den Peer-Review zu erstellen.

CTAN:macros/latex/contrib/anonymous-acm

unitconv von *Herbert Voß* stellt zwei Makros bereit, mit denen ein Wert aus einer Einheit in den entsprechenden Wert aus einer anderen Einheit konvertiert werden kann. Unterstützt werden die bekannten $\text{T}\text{\TeX}$ -Längeneinheiten sowie Kilometer und Meter. Das Paket funktioniert nur mit $\text{Lua}\text{\LaTeX}$.

CTAN:macros/luatex/latex/unitconv

metanorma von *Paolo Brasolin* ist eine Sammlung aus Klasse, Skript und Stylesheet, mit dem man eine Reihe von Standardisierungs-Dokumenten mit dem Framework Metanorma verfassen kann.

CTAN:macros/latex/contrib/metanorma

jlreq-deluxe von *Yukimasa Morimi* bietet eine Unterstützung für verschiedene Schriftarten zur Verwendung mit der Dokumentenklasse *jlreq* von *Noriyuki Abe*.

CTAN:language/japanese/jlreq-deluxe

biblatex-software von *Roberto Di Cosmo* ist ein Biblatex-Stil, mit dem man Software zitieren kann.

CTAN:macros/latex/contrib/biblatex-contrib/biblatex-software

epigraph-keys von *Benjamin McKay* dient dazu, kurze Zitate und Sinnsprüche am Anfang oder am Ende eines Kapitels zu setzen.

CTAN:macros/latex/contrib/epigraph-keys

willowtreebook von *Benjamin McKay* stellt ein Interface zu der Dokumentenklasse *memoir* bereit, um dessen Verwendung zu erleichtern. Der Autor verwendet die Klasse für seine Vorlesungsmitschriften.

CTAN:macros/latex/contrib/willowtreebook

mercatormap von *Thomas F. Sturm* erweitert TikZ und stellt ein Pythonskript zur Verfügung, mit dem man auf freie Landkarten zugreifen kann, um sie in Dokumenten zu verwenden.

CTAN:macros/latex/contrib/mercatormap

diabetes-logbook von *Viktor Grigorov* ist ein Paket, mit dem Typ-1-Diabetiker, die sich mit einer intensivierten konventionellen Insulintherapie (ICT) behandeln,

ihr Diabetestagebuch zur Dokumentation ihrer Stoffwechsellage führen können.

CTAN:macros/latex/contrib/diabetes-logbook

dpcircling von *Oura Mitsuaki* erzeugt vier Arten von Umrandungen für Fließtext, um einzelne Wörter grafisch hervorzuheben. Verfügbar sind ein Oval, ein Rechteck, ein angerissenes Rechteck und eine scheibenwischerartige Form.

CTAN:graphics/pgf/contrib/dpcircling

ccool von *Erwann Rogard* ist der Nachfolger des Pakets *oops* und nicht zu verwechseln mit dem Paket *cool* von *Nick Setzer*. Bereitgestellt wird ein minimalistisches Interface für mathematische Ausdrücke.

CTAN:macros/latex/contrib/ccool

emojicite von *Leon Sixt* bereichert Fundstellen in wissenschaftlichen Arbeiten um Emojis. Dazu werden das Paket *emoji* von *Xiangdong Zeng* und *Lua^AT_EX* benötigt.

CTAN:macros/luatex/latex/emojicite

endnotes-hy von *Donald P. Story* behebt einen Fehler im Paket *endnotes*, so dass Hyperlinks zu den korrekten Ankern erzeugt werden.

CTAN:macros/latex/contrib/endnotes-hy

Spielplan

2020

22. 8. – 23. 8. **FrOSCon**, Free and Open Source Conference (Online)
Hochschule Bonn-Rhein-Sieg
Grantham-Allee 20, 53757 Sankt Augustin
<https://www.froscon.de/startseite/>



6. 9. – 12. 9. **14th International ConT_EXt Meeting**
Prag-Sibřina, Tschechische Republik
<https://meeting.contextgarden.net/>



18. 9. – 19. 9. **18. Kieler Open Source und Linux Tage** (Online)
Kitz – Kieler Innovations- und Technologiezentrum
Schauenburgerstr. 116
24118 Kiel
<https://www.kieler-linuxtage.de>



2021

10. 3. – 12. 3. **DANTE 2021**
und 64. Mitgliederversammlung von DANTE e.V.
Otto-von-Guericke-Universität
39016 Magdeburg
<https://www.dante.de/veranstaltungen/dante2021/>



Stammtische



In verschiedenen Städten im Einzugsbereich von DANTE e.V. finden regelmäßig Treffen von T_EX-Anwendern statt, die für jeden offen sind. Im Web gibt es aktuelle Informationen unter <https://projekte.dante.de/Stammtische/WebHome>.

Aachen

Torsten Bronger,
bronger@physik.rwth-aachen.de
Restaurant Galilei, Marienbongard 24, 52062 Aachen
Erster Donnerstag im Monat, 20.00 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

Erlangen

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



Göttingen

Holger Nobach,
holger.nobach@nambis.de, <http://goetex.nambis.de/>
Restaurant Mazzoni Cucina Italiana,
Hermann-Rein-Straße 2, 37075 Göttingen
Dritter Donnerstag im Monat, 18.00 Uhr

Hamburg

Günther Zander,
guenther.zander@lug-balista.de
Bürgerhaus in Barmbek, Lorichsstraße 28a, 22307 Hamburg
Zweiter Montag im Monat, 20.00 Uhr

Hannover

Reik Kaps,
kaps@luis.uni-hannover.de
Raum B024 im EG des LUIS,
Schloßwender Str. 5, 30419 Hannover
Zweiter Donnerstag im Monat, 18.30 Uhr

Heidelberg

Martin Wilhelm Leidig, Tel.: 0170 418 33 29,
moss@moss.in-berlin.de
Anmeldeseite zur Mailingliste: <https://tinyurl.com/stammtisch-HD>
Wechselder Ort
Letzter Freitag im Monat, ab 19.30 Uhr



Köln

Uwe Ziegenhagen
Dingfabrik, Erzbergerplatz 9, 50733 Köln
zur Zeit inaktiv, 19.00 Uhr

Stuttgart

Bernd Raichle,

bernd.raichle@gmx.de

»Trollinger-Stubn«, Rotebühlstr. 50, 70178 Stuttgart

Zweiter Dienstag im Monat, 19.30 Uhr

Trier

Martin Sievers,

tex-stammtisch-trier@schoenerpublizieren.de

Universität Trier

nach Vereinbarung

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

Adressen

DANTE, Deutschsprachige Anwendervereinigung T_EX e.V.

Postfach 11 03 61

69072 Heidelberg

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

Fax: (0 62 21) 16 79 06

E-Mail: info@dante.de

Konto: VR Bank Rhein-Neckar eG

IBAN DE67 6709 0000 0002 3100 07 SWIFT-BIC GENODE61MA2

Vorstand

Vorsitzender:	Martin Sievers	president@dante.de
stv. Vorsitzender:	Herbert Voß	vice-president@dante.de
Schatzmeisterin:	Doris Behrendt	treasurer@dante.de
Schriftführer:	Manfred Lotz	secretary@dante.de
Beisitzer:	Klaus Höppner	
	Harald König	
	Volker RW Schaa	
	Uwe Ziegenhagen	

Ehrenmitglieder

Peter Sandner	22.03.1990	Klaus Thull († 2012)	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 († 2015)	19.02.1999
Joachim Lammarsch	12.04.2014	Rainer Schöpf	12.04.2014

Webserver und Mailingliste

DANTE: <https://www.dante.de/> (Erik Braun)
CTAN: <https://mirror.ctan.org/> (Gerd Neugebauer)
DANTE-EV: <https://lists.dante.de/mailman/listinfo/dante-ev>

FAQ

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

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

T_EXnische Fragen

beraterkreis@dante.de

ak-schule@dante.de

Autoren/Organisatoren

Doris Behrendt doris@dante.de	[11,23]	Stephan Lukasczyk Reisacher Weg 1a 94036 Passau tex@lukasczyk.me	[6]
Pascal Braband info@pascalbraband.de	[48]		
Jürgen Fenn Christian-Stock-Straße 4 63263 Neu-Isenburg juergen.fenn@gmx.de	[58]	Marei Peischl marei@peischl.name	[6]
Mario Haustein Mario.Haustein@hrz.tu-chemnitz.de	[6,23]	Nils Pickert nils@mipi.de	[23]
Johannes Hielscher jhielscher@posteo.de	[23]	Henning Hraban Ramm hraban@fieee.net	[23]
Alexander Krumeich n-design GmbH Alpenerstr. 16 50825 Köln alexander.krumeich@n-design.de	[37]	Martin Sievers siehe Seite 66	[4]
		Herbert Voß Wasgenstraße 21 14129 Berlin herbert@dante.de	[3,51]

Die T_EXnische Komödie

32. Jahrgang Heft 3/2020 August 2020

Impressum

Editorial

Hinter der Bühne

- 4 Grußwort
- 6 Bericht der Rechnungsprüfer zum Vereinsjahr 2019
- 11 Vorläufiger Bericht der Schatzmeisterin für das Jahr 2019
- 23 CCCamp19

Bretter, die die Welt bedeuten

- 37 n-doc – ein L^AT_EX-basiertes Verfahren für IT-Sicherheitszertifizierungen
- 48 Professionell präsentieren mit der Beamer-Klasse
- 51 Chaotische Symmetrien mit Lua berechnet

Von fremden Bühnen

- 58 Neue Pakete auf CTAN

Spielplan

- 63 Termine
- 64 Stammtische

Adressen

- 67 Autoren/Organisatoren