

Die T_EXnische Komödie

dante

Deutschsprachige
Anwendervereinigung T_EX e.V.

35. Jahrgang Heft 2/2023 Mai 2023

2/2023

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 1.16.0 (TeX Live 2023) erstellt. Als Standardschriften kamen Libertinus Serif, Libertinus Sans Serif, Anonymous Pro und Libertinus Math zum Einsatz.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 2150

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

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

Druck: Schleunungsdruck GmbH
Eltertstraße 27, 97828 Marktheidenfeld

Redaktion: Luzia Dietsche (verantwortliche Redakteurin)

Mitarbeit: Adelheid Bonnetsmüller Eberhard Lisse Ralf Mispelhorn
Rolf Niepraschk Stefan Pinnow Bernd Raichle
Christine Römer Herbert Voß

Redaktionsschluss für Heft 3/2023: 15. Juli 2023

ISSN 1434-5897

Die T_EXnische Komödie 2/2023

Editorial

Liebe Leserinnen und Leser,

auch dies Mal haben fleißige Autoren dazu beigetragen, dass eine abwechslungsreiche Ausgabe unserer Mitgliederzeitung zustande kam. So ist das Thema $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ durch die Artikel zur Schulphysik und zu CD-Cover abgedeckt, $\text{ConT}_{\text{E}}\text{Xt}$ durch die Beiträge zu Kalendern und $\text{ConT}_{\text{E}}\text{Xt}$ -Neuigkeiten und $\text{T}_{\text{E}}\text{X}$ zumindest indirekt durch den Artikel über Donald Knuth vertreten. Es ist für mich zutiefst beeindruckend, was dieser Mann alles geschaffen hat. Ob es wohl auch mal wieder einen Beitrag zu METAFONT geben wird? Oder Grafikeinbindung? Oder Fußnotenapparate? Wovon eine Chefredakteurin in der Nacht alles träumt...

Allerdings sollte ich meine Träume vielleicht etwas zügeln – wer weiß, ob ich auch für solche Herausforderungen die Hilfe finden würde, die ich für jede Ausgabe benötige. Ein großes Dankeschön an all die fleißigen Helferlein, ohne die ich keine einzige Ausgabe bisher erfolgreich gemeistert hätte. Jede, aber auch wirklich jede Ausgabe hat mich mit irgendetwas an den Rand der Verzweiflung gebracht, von dem mich der eine oder die andere gerettet hat. Vielen Dank!

Ich freue mich immer wieder, wenn wir in der Redaktion Hinweise auf interessante, unsere Themen betreffende Ausstellungen, Bücher oder Web-Seiten erhalten. Diese Hinweise dürfen gerne als mehrseitige Besprechungen eingehen, können aber genauso gut in einer Kurzzusammenfassung oder nur der einfachen Nennung eines Links bestehen.

So habe ich mit großem Interesse den Link <https://ilovetypography.com/2014/10/15/the-first-female-typographer/> aufgerufen und über frühe Typographinnen nachgelesen. Eigentlich ist es zu erwarten, dass Nonnen genauso wie Mönche schon früh (schon immer?) Schriften und Bücher abgeschrieben haben, aber mir war diese Tatsache nicht bewusst. Beeindruckend finde ich, dass es auch im Mittelalter mutige Frauen gab, die sich im Bereich Bücher, (Ab)Schreiben und Druck ihr Tätigkeitsfeld suchten. Danke für den Hinweis!



Und damit wünsche ich Ihnen und Euch wieder viel Spaß beim Lesen und verbleibe mit $\text{T}_{\text{E}}\text{X}$ nischen Grüßen

Luzia Dietsche



Hinter der Bühne

Vereinsinternes

Grußwort

Liebe Mitglieder,

der Mai ist gekommen und mit ihm erstes sommerliches Wetter. Der Wonnemonat bringt daneben seit vielen Jahren schon die T_EX Collection. So findet Ihr auch diesmal eine DVD im Briefumschlag, die den aktuellen Zustand des T_EX-Universums festhält.

Aufgrund des eingespielten Prozesses haben wir trotz fehlender Bürokratie auch dieses Jahr wieder die Produktion und den Versand der DVDs für die T_EX-Nutzergruppen weltweit übernommen. Ein besonderer Dank geht dabei an Karl Berry, Manfred Lotz und Klaus Höppner für die Zusammenstellung der Collection und die Erstellung des Masters, an Herbert Voß für das Cover sowie an Doris Behrendt für die Bestellabwicklung und den Versand.

Der Mai bzw. das Grußwort in der jeweils zweiten Jahresausgabe der Mitgliederzeitung war früher die Gelegenheit, auf die Frühjahrstagung zurückzublicken, über Veränderungen im Vorstand zu berichten etc. In diesem Jahr haben wir uns, wie bereits im vergangenen Grußwort angekündigt, dafür entschieden, die diesjährige Mitgliederversammlung am 13. Juli unmittelbar vor der Tagung TUG 2023 zu veranstalten. Die offizielle Einladung ist auf S. 5 zu finden. Es stehen u. a. Neuwahlen zum Vorstand auf dem Programm. Für eine bessere Planbarkeit bitte ich um frühestmögliche Anmeldung über <https://www.dante.de/veranstaltungen/dante2023/anmeldung/>.

Erfreulicherweise hat der Gesetzgeber mittlerweile das Vereinsrecht dahingehend angepasst, dass wir in Zukunft auch ohne Satzungsänderung hybride oder sogar virtuelle Mitgliederversammlungen einberufen können. Damit steht einer digitalen Komponente der Frühjahrstagung 2024 in Oßmannstedt von dieser Seite nichts im Wege.

Doch das ist noch eine Weile hin. Zunächst wünsche ich viel Vergnügen bei der weiteren Lektüre dieser Ausgabe der Mitgliederzeitung.

Herzlichst Ihr/Euer
Martin Sievers

Einladung zur 65. Mitgliederversammlung von DANTE e.V. am 13. Juli 2023 in Bonn

Martin Sievers

Liebe Mitglieder von DANTE e.V.,

ich lade Sie/Euch auch im Namen des Vorstands ganz herzlich zur 65. Mitgliederversammlung am 13. Juli 2023 von 10.00 bis 13.00 Uhr in Bonn ein.

Die Anschrift lautet:

Hotel Collegium Leoninum
Kapitelsaal
Noeggerathstraße 34
53111 Bonn

Die Tagesordnung sieht wie folgt aus:

1. Begrüßung und Tagesordnung
2. Bericht des Vorstands
3. Finanzbericht
4. Bericht der Rechnungsprüfer
5. Entlastung des Vorstands
6. Wahl eines Vorstands
7. Wahl von Rechnungsprüfern
8. Verschiedenes

Anträge zur Ergänzung oder Änderung der Tagesordnung sind gemäß § 12 (3) der Vereinssatzung bis zum 15. Juni schriftlich an den Vorstand zu stellen.

Ihre Stimmunterlagen erhalten Sie direkt vor Ort; um vorherige Anmeldung unter <https://www.dante.de/veranstaltungen/dante2023/anmeldung/> wird herzlich gebeten. Eine Übertragung des Stimmrechts ist im Rahmen des § 13 (4) der Vereinssatzung möglich. Wie üblich sind auch Nichtmitglieder als Gäste herzlich willkommen.

Fragen, Wünsche und Anregungen können am besten direkt an mv65@dante.de gerichtet werden.

Mit freundlichen Grüßen

Martin Sievers (Vorsitzender DANTE e.V.)



Einladung zum bayrischen T_EX-Stammtisch BayT_EX 2023

Oliver Rath

Liebe T_EX-Begeisterte,

ob bayrisch oder anderer Nation, wir freuen uns, auch dieses Jahr wieder einen bayrischen T_EX-Stammtisch BayT_EX anbieten zu können.

28. 7. – 29. 7. 2023 BayT_EX 2023
bei der Genua GmbH
Domagkstraße 7
85551 Kirchheim bei München
<https://baytex.dante.de>



Vorläufiges Programm

Freitag, den 28.7.

ab 9.30 Uhr Anreise, Empfang in die Domagkstraße 7 in Kirchheim. Anmeldung erforderlich (Infos weiter hinten)

10.00–10.15 Uhr Grußwort des Geschäftsführers Matthias Ochs

10.15–13.00 Uhr Vorträge

13.00–14.00 Uhr Mittagessen in der Genua Aula

14.00–17.00 Uhr Vorträge

ab 18.00 Uhr Führung und Verkostung mit Brotzeit in der Zehmer Bräu Kirchheim

Samstag, den 29.7.

ab 9.30 Uhr Einlass

10.00–13.00 Uhr Vorträge

ab 13.00 Uhr Grillen im Genua Garten

ab 15.00 Uhr Gin Tasting bei Dukes Gin Distillery (angefragt)

Weitere Informationen

Eine Anmeldung ist obligatorisch über die Email-Adresse eventmanagement@genua.de. Wir benötigen den vollständigen Namen und eventuell Firmennamen. Außerdem

bitte angeben, dass die Anmeldung für die BayT_EX 2023 sein soll. Da Genua ISO-zertifiziert ist, kommen nur angemeldete Leute rein.

Call for Papers

Wir sind immer noch auf der Suche nach spannenden, anregenden, aufregenden und/oder faszinierenden Vorträgen für die beiden Tage und freuen uns auf Ihre aktive Teilnahme. Alles, was sich um T_EX und Textsatz auch im weiteren Sinne dreht, ist willkommen, mögliche Themen eines Beitrags können sein:

- Erfahrungsberichte zum Einsatz von T_EX bzw. Open-Source-Software,
- Einsatz von T_EX in Lehre und Forschung an wissenschaftlichen Einrichtungen,
- Nutzung von T_EX für den Satz von Facharbeiten bzw. anderen Abschlussarbeiten, Präsentationen etc. an (Hoch-)Schulen,
- Vorstellung spezieller Erweiterungen für den Einsatz in Beruf und Ausbildung,
- Einführungen in T_EX und die zugehörigen Makropakete (L^AT_EX 2_ε/L^AT_EX 3, ConT_EXt, ...),
- Lösungen mit den neueren Engines X_YT_EX und LuaT_EX,
- Beispiele aus der Praxis (beispielsweise Realisierung besonderer Anforderungen), eigene Klassen und Pakete,
- Einbinden von Schriften, Grafiken etc.,
- Typografie und ihre Umsetzung in T_EX und Co.,
- Zusammenspiel von T_EX mit anderen Dateiformaten (z. B. XML) und anderen (Open-Source-)Werkzeugen,
- die Entwicklung von T_EX und Co. in den vergangenen Jahrzehnten,
- barrierefreie PDF-Dokumente,
- ...

Fühlen Sie sich angesprochen? Dann senden Sie bitte eine E-Mail an oliver_rath@genua.de.

Bretter, die die Welt bedeuten

L^AT_EX und Schulphysik 2: Schaltbilder

Keno Wehr

Der zweite Teil der Artikelreihe zur Schulphysik erläutert das Zeichnen von Schaltbildern mit den Paketen `pst-circ`, `circuitikz`, `circuits.ee.IEC` und `schulma-physik`. Außerdem werden die in der Schule üblichen Konventionen für Schaltzeichen zusammengestellt.

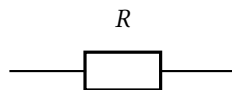
Zum Zeichnen von Schaltbildern gibt es eine ganze Reihe von Paketen (siehe CTAN-Thema »`diagram-circ`«) auf der Grundlage unterschiedlicher Grafiksprachen. Hier sollen ein Paket auf der Grundlage von PSTricks und zwei auf der Grundlage von TikZ vorgestellt werden. Ein Überblick über die speziellen Bedürfnisse der Schulphysik führt dann zu einigen Ergänzungen durch das Paket `schulma-physik`. Die voreingestellte Längeneinheit für Koordinaten ist in allen hier aufgeführten Paketen 1 cm.

Das Paket `pst-circ`

`pst-circ` (vgl. auch [1]) basiert auf PSTricks. Daher sollten Dokumente, die dieses Paket verwenden, mit `lualatex` übersetzt werden.

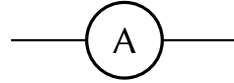
In `pst-circ` wird jedes Element des Schaltkreises auf der Grundlage von zwei oder mehr Punkten, die es verbindet, festgelegt. Im folgenden Beispiel werden zunächst zwei Punkte A und B definiert und dann durch einen Widerstand verbunden.

```
\pnode(0,0){A}  
\pnode(3,0){B}  
\resistor(A)(B){R$}
```



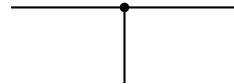
Entsprechend können viele weitere Schaltelemente erzeugt werden, die der Paketanleitung zu entnehmen sind. Für Messinstrumente eignet sich das generische Element `\circledipole` mit der Option `labeloffset=0`.

```
\node(0,0){A}
\node(3,0){B}
\circledipole[labeloffset=0](A)(B){\Large\textsf
  ↪{A}}
```



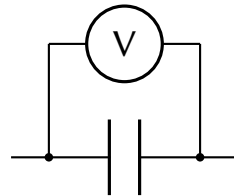
Reine Kabelstrecken werden mit `wire` dargestellt. Die Option `arrows` ermöglicht die Darstellung von Verzweigungspunkten. Der Befehl `\pnodes` erlaubt die gleichzeitige Definition mehrerer Punkte.

```
\pnodes(0,0){A}(3,0){B}
\wire(A)(B)
\pnodes(1.5,0){C}(1.5,-1){D}
\wire[arrows=*-](C)(D)
```



Schaltelemente, die parallel zu anderen geschaltet sind, können mit der Option `parallel` bequem dargestellt werden. Die zusätzliche Option `parallelnode` sorgt dafür, dass auch die nötigen Kontaktpunkte gesetzt werden.

```
\pnodes(0,0){A}(3,0){B}
\capacitor(A)(B){}
\pnodes(0.5,0){C}(2.5,0){D}
\circledipole[labeloffset=0,parallel,
  ↪parallelnode](C)(D){\Large\textsf{V}}
```



Wir verbinden nun die Elemente zu einem Schwingkreis und zu einer Parallelschaltung von Glühlampen (Abb. 1). Der gesamte Code für ein Schaltbild wird in die Umgebung `pspicture` eingeschlossen, der die Koordinaten der linken unteren und der rechten oberen Ecke des Bildes übergeben werden. Wenn die Koordinaten zu knapp bemessen sind, kann es zu Überschneidungen mit dem umgebenden Text kommen.

```
\begin{pspicture}(-0.2,-2.9)(5.4,2.9)
\nodes(0,-2){A}(0,2){B}
\OpenDipol[labeloffset=0](A)(B){\large$\sim$}
\nodes(4.5,2){C}(4.5,-2){D}(2,-2){E}
\coil(B)(C){\mathit{L}}
\capacitor(C)(D){\mathit{C}}
\resistor(D)(E){\mathit{R}}
\circledipole[labeloffset=0](E)(A){\Large\textsf{A}}
\end{pspicture}
\hspace{1em}
\begin{pspicture}(-2.9,-0.6)(2.4,6.9)
\nodes(-2.3,0){A}(2.3,0){B}
```

```

\battery(A)(B){}
\nodes(-2.3,1.8){C}(2.3,1.8){D}
\lamp[arrows=**](C)(D){}
\nodes(-2.3,4.8){E}(2.3,4.8){F}
\lamp(E)(F){}
\circledipole[labeloffset=0](A)(C){\Large\textsf{A}}
\circledipole[labeloffset=0](C)(E){\Large\textsf{A}}
\wire(B)(F)
\nodes(-1.2,1.8){G}(1.2,1.8){H}
\circledipole[parallel,parallelnode,labeloffset=0](G)(H){\Large\textsf{V}}
\nodes(-1.2,4.8){I}(1.2,4.8){J}
\circledipole[parallel,parallelnode,labeloffset=0](I)(J){\Large\textsf{V}}
\end{pspicture}

```

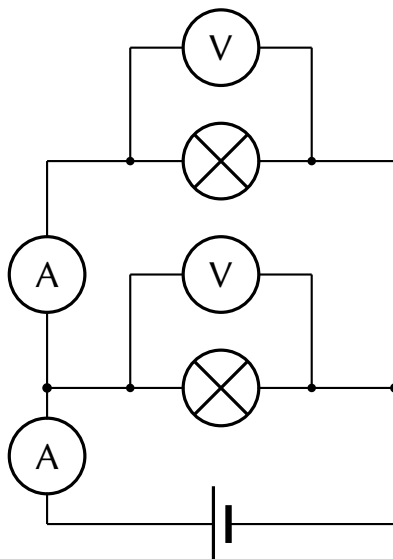
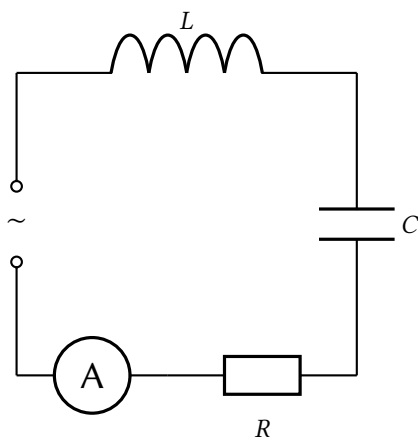


Abb. 1: Schaltbild eines Schwingkreises und einer Parallelschaltung von Glühlampen mit `pst-circ`

Das Paket `circuitikz`

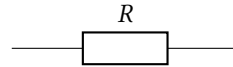
Die wichtigste Alternative zu `pst-circ` ist das Paket `circuitikz`, das jenes in der Anzahl der zur Verfügung stehenden Schaltelemente und den Konfigurationsmöglichkeiten noch übersteigt. Für die meisten Elemente gilt ein ähnliches Zeichenprinzip wie in `pst-circ`: Man verbindet zwei festgelegte Punkte durch ein Schaltelement.

```
\begin{circuitikz}
\draw (0,0) to[resistor={R$}] (3,0);
\end{circuitikz}
```



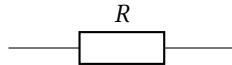
Das auf diese Weise erzeugte Schaltbild des Widerstands ist in deutschsprachigen und anderen europäischen Ländern unüblich. Abhilfe schafft die Verwendung der Option `resistor=european`, die mit dem Befehl `\ctikzset` gewählt werden kann.

```
\ctikzset{resistor=european}
\begin{circuitikz}
\draw (0,0) to[resistor={R$}] (3,0);
\end{circuitikz}
```



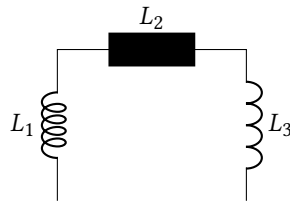
Alternativ ist es möglich, die europäische Form ausdrücklich mit `european resistor` anzufordern.

```
\begin{circuitikz}
\draw (0,0) to[european resistor={R$}] (3,0);
\end{circuitikz}
```



Für Spulen stehen gleich drei verschiedene Schaltzeichen zur Verfügung.

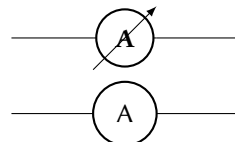
```
\begin{circuitikz}
\draw (0,0) to[inductor={L_1$}] (0,2)
to[european inductor={L_2$}] (2.5,2)
to[american inductor={L_3$}] (2.5,0);
\end{circuitikz}
```



In deutschsprachigen Schulbüchern findet generell die »amerikanische« Form Verwendung. Um diese als Standardform einzustellen, übergibt man dem Befehl `\ctikzset` die Option `inductor=american`.

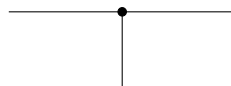
Für Stromstärke- und Spannungsmessgeräte stehen Standardelemente (`ammeter` und `voltmeter`) zur Verfügung, die eine in der Schule unübliche Form haben. Man kann sich mit dem generischen Element `rmeter` behelfen.

```
\begin{circuitikz}
\draw (0,1) to[ammeter] (3,1);
\draw (0,0) to[rmeter,t={\textsf{A}}] (3,0);
\end{circuitikz}
```



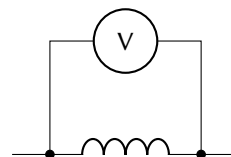
Bei reinen Kabelstrecken kann man zwei Bindestriche – den Standard-TikZ-Befehl für gerade Linien – zum Verbinden der Punkte verwenden. Falls am Ende Kontaktpunkte liegen, ist stattdessen der Verbindungstyp `short` mit einer der Zusatzoptionen `*`, `—` oder `—*` zu wählen.

```
\begin{circuitikz}
\draw (0,0)--(3,0);
\draw (1.5,0) to[short,*-] (1.5,-1);
\end{circuitikz}
```



Wie in pst-circ lassen sich auch vorab Punkte definieren und dann für die Zeichnung verwenden.

```
\ctikzset{inductor=american}
\begin{circuitikz}
\coordinate (A) at (0,0);
\coordinate (B) at (3,0);
\draw (A) to[inductor] (B);
\coordinate (C) at (0.5,0);
\coordinate (D) at (2.5,0);
\coordinate (E) at (0.5,1.5);
\coordinate (F) at (2.5,1.5);
\draw (C) to[short,*-] (E) to[rmeter,t={\textsf{
\hookrightarrow V}}] (F) to[short,*-] (D);
\end{circuitikz}
```



Zum Vergleich zeichnen wir auch mit circuitikz einen Schwingkreis und eine Parallelschaltung von Glühlampen (Abb. 2).

```
\ctikzset{resistor=european,inductor=american}
\begin{circuitikz}
\draw (0,0.4) to[short,o-] (0,2) to[inductor={\mathbb{L}}] (4.5,2)
to[capacitor={\mathbb{C}}] (4.5,-2) to[resistor={\mathbb{R}}] (2,-2)
to[rmeter,t={\textsf{A}}] (0,-2) to[short,o-] (0,-0.4);
\node at (0,0) {\mathbb{sim}};
\end{circuitikz}
\hspace{1em}
\begin{circuitikz}
\coordinate (A) at (-2.3,0); \coordinate (B) at (2.3,0);
\coordinate (C) at (-2.3,1.8); \coordinate (D) at (2.3,1.8);
\coordinate (E) at (-2.3,4.8); \coordinate (F) at (2.3,4.8);
\draw (A) to[rmeter,t={\textsf{A}}] (C) to[rmeter,t={\textsf{A}}] (E)
to[lamp] (F) to (B) to[battery1] (A);
\draw (C) to[lamp,*-] (D);
\coordinate (G) at (-1.3,1.8); \coordinate (H) at (1.3,1.8);
\coordinate (I) at (-1.3,3.3); \coordinate (J) at (1.3,3.3);
\draw (G) to[short,*-] (I) to[rmeter,t={\textsf{V}}] (J) to[short,*-] (H);
\coordinate (K) at (-1.3,4.8); \coordinate (L) at (1.3,4.8);
\coordinate (M) at (-1.3,6.3); \coordinate (N) at (1.3,6.3);
```

```
\draw (K) to[short,*-] (M) to[rmeter,t={\textsf{V}}] (N) to[short,*-] (L);
\end{circuitikz}
```

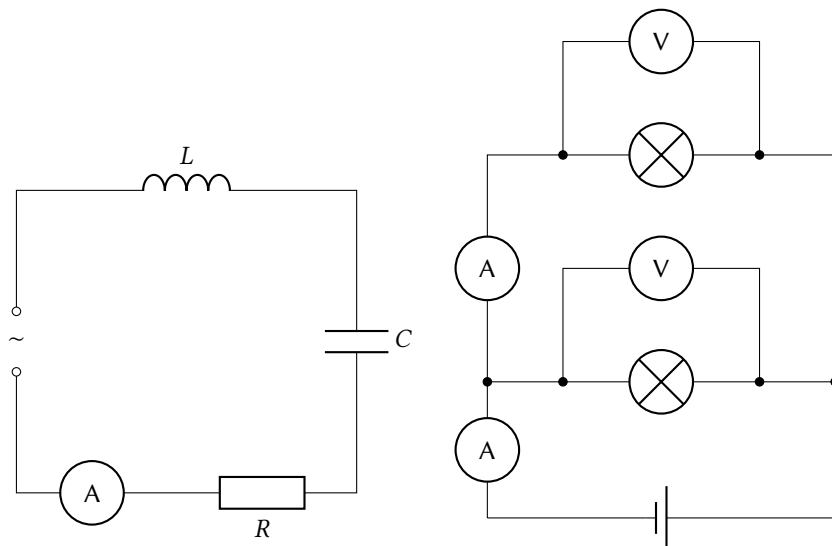


Abb. 2: Schaltbild eines Schwingkreises und einer Parallelschaltung von Glühlampen mit circuitikz

Die TikZ-Bibliothek `circuits.ee.IEC`

Neben dem gerade vorgestellten `circuitikz`, das auf `TikZ` basiert, aber als eigenständiges Paket geführt wird, gibt es in `TikZ` selbst die Bibliothek `circuits.ee.IEC` mit der gleichen Zielsetzung. Dabei steht `ee` für *electrical engineering* (in Abgrenzung zu der ebenfalls existierenden Bibliothek für Logikschaltkreise) und `IEC` für die Normungsorganisation *International Electrotechnical Commission*.

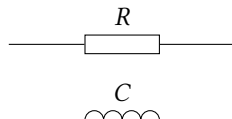
Die Syntax dieser Bibliothek ist der von `circuitikz` wenig überraschend recht ähnlich, doch nimmt sich der Umfang des angebotenen Schaltzeichenkatalogs deutlich bescheidener aus. Für viele Zwecke sind die vorhandenen Zeichen aber ausreichend und gewisse Ergänzungen können mit gewöhnlichen `TikZ`-Befehlen vorgenommen werden, weshalb diese Bibliothek hier zusätzlich vorgestellt wird. Sie wird folgendermaßen geladen:

```
\usetikzlibrary{circuits.ee.IEC}
```

Zu beachten ist, dass `circuits.ee.IEC` und `circuitikz` untereinander inkompatibel sind, also nicht in einem Dokument nebeneinander verwandt werden können.¹

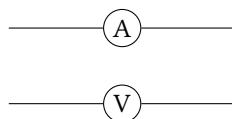
Die für die Schulphysik zu bevorzugenden Formen der Schaltzeichen für Widerstand und Spule entsprechen in `circuits.ee.IEC` bereits der Standardeinstellung. Die Beschriftung der Schaltzeichen erfolgt mit der Option `info`. Die Bildumgebung `tikzpicture` muss mit der Option `circuit ee IEC` (hier ohne `s` und ohne Punkte!) aufgerufen werden.

```
\begin{tikzpicture}[circuit ee IEC]
\draw (0,1) to[resistor={info={R$}}] (3,1);
\draw (0,0) to[inductor={info={C$}}] (3,0);
\end{tikzpicture}
```



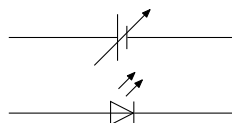
Spannungs- und Stromstärkemessgeräte in geläufiger Form lassen sich etwas einfacher erzeugen als mit den zuvor vorgestellten Paketen.

```
\begin{tikzpicture}[circuit ee IEC]
\draw (0,1) to[amperemeter] (3,1);
\draw (0,0) to[voltmeter] (3,0);
\end{tikzpicture}
```



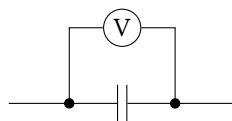
Hilfreich, um bestimmte Schaltelemente mit zusätzlichen Informationen auszustatten, sind die sogenannten Annotationen `adjustable` und `light emitting`.

```
\begin{tikzpicture}[circuit ee IEC]
\draw (0,1) to[battery=adjustable] (3,1);
\draw (0,0) to[diode=light emitting] (3,0);
\end{tikzpicture}
```



Um Verzweigungspunkte anzulegen, verwendet man am besten die aus TikZ bekannte `\node`-Syntax mit dem Schaltelement `contact` als Knotentyp.

```
\begin{tikzpicture}[circuit ee IEC]
\draw (0,0) to[capacitor] (3,0);
\node[contact] (A) at (0.8,0) {};
\node[contact] (B) at (2.2,0) {};
\draw (A)--(0.8,1) to[voltmeter] (2.2,1)--(B);
\end{tikzpicture}
```



Nun zeichnen wir wieder einen Schwingkreis und eine Parallelschaltung von Glühlampen (Abb. 3). Der in `circuits.ee.IEC` fehlende offene Kreis für eine Buchse als Teil einer Spannungsquelle wird durch einen gewöhnlichen TikZ-Knoten mit Kreisform und Größe von 3 pt realisiert.

¹`circuitikz` hat allerdings eine Option `compatibility`, die die Vereinbarkeit mit `circuits.ee.IEC` sicherstellen soll. Details sind in der Paketanleitung nachzulesen.

```

\begin{tikzpicture}[circuit ee IEC]
\node[circle,draw,inner sep=0pt,minimum size=3pt] (A) at (0,0.3) {};
\node[circle,draw,inner sep=0pt,minimum size=3pt] (B) at (0,-0.3) {};
\draw (A)--(0,1.5) to[inductor={info={\$L\$}}] (3.6,1.5)
to[capacitor={info={\$C\$}}] (3.6,-1.5) to[resistor={info={\$R\$}}] (1.8,-1.5)
to[amperemeter] (0,-1.5)--(B);
\node at (0,0) {\$sim\$};
\end{tikzpicture}
\hspace{2em}
\begin{tikzpicture}[circuit ee IEC]
\coordinate (A) at (-2,0); \coordinate (B) at (2,0);
\node[contact] (C) at (-2,1.5) {}; \node[contact] (D) at (2,1.5) {};
\coordinate (E) at (-2,4); \coordinate (F) at (2,4);
\draw (A) to[amperemeter] (C) to[amperemeter] (E) to[bulb] (F)--(B)
to[battery] (A);
\draw (C) to[bulb,*-] (D);
\node[contact] (G) at (-1,1.5) {}; \node[contact] (H) at (1,1.5) {};
\coordinate (I) at (-1,2.5); \coordinate (J) at (1,2.5);
\draw (G)--(I) to[voltmeter] (J)--(H);
\node[contact] (K) at (-1,4) {}; \node[contact] (L) at (1,4) {};
\coordinate (M) at (-1,5); \coordinate (N) at (1,5);
\draw (K)--(M) to[voltmeter] (N)--(L);
\end{tikzpicture}

```

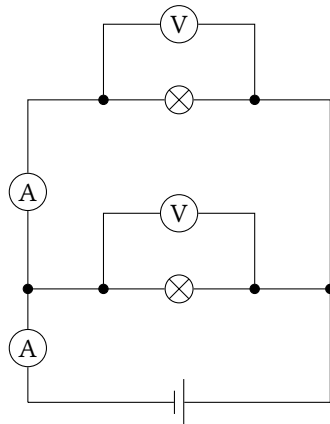
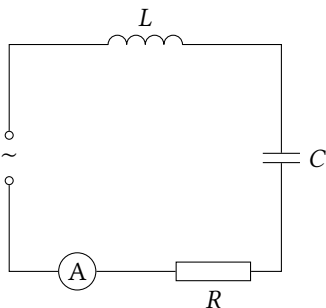


Abb. 3: Schaltbild eines Schwingkreises und einer Parallelschaltung von Glühlampen mit circuits.ee.IEC

Schaltzeichen in der Schulphysik

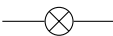
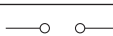
Das Aussehen von Schaltzeichen ist weniger einheitlich, als man meinen könnte. Neben nationalen Traditionen spielen die Gepflogenheiten des jeweiligen Arbeitsgebietes und nicht zuletzt der persönliche Geschmack des Autors eine Rolle bei der Gestaltung von Schaltbildern. Uns interessieren hier nun die (ungeschriebenen) Konventionen der Schulphysik und ihre Umsetzung mit L^AT_EX.

Für Widerstand und Spule sind in deutschsprachigen Schulbüchern die Formen  und  allgemein üblich. In `pst-circ` und `circuits.ee.IEC` sind dies die Standardformen; für `circuitikz` wurden oben die Optionen genannt, mit denen diese Formen ausgewählt werden können. In leichter Abwandlung der Hauptform werden Spulen teilweise auch mit nur drei oder mehr als vier Bögen dargestellt. Zum Teil geschieht dies aus Inkonsequenz, zum Teil ist erkennbar, dass hiermit eine Spule mit größerer Windungszahl veranschaulicht werden soll. Wer dies nachahmen will, muss ein eigenes Schaltzeichen entwerfen, da unsere Pakete es nicht unterstützen. Im Allgemeinen dürfte es jedoch sinnvoller sein, auf solche Variationen zu verzichten und nötige Kenndaten wie die Windungszahl im Klartext anzugeben.

Für Dioden verwenden die Schulbücher durchweg  (nicht  oder ). Diese Form ist in `circuits.ee.IEC` der Standard für das Element `diode`. In `circuitikz` ist sie als `stroke diode` verfügbar, während `pst-circ` sie nicht anbietet.

Zeichen wie  und  für Spannungsquellen sind in der Schulphysik gänzlich ungebräuchlich. Für Wechselspannungsquellen wird immer  verwendet, oft mit einer Schlangenlinie zwischen oder neben den beiden Buchsen. Für Gleichspannungsquellen ist der Brauch weniger klar. Oft findet man , was offiziell für eine Batterie steht, auch an Stellen, wo der experimentelle Einsatz einer Batterie unpraktikabel ist, so bei Beschleunigungsspannungen im Kilovoltbereich oder Versuchen, die eine Variation der Spannung erfordern (dann gelegentlich durch  verdeutlicht). Andererseits findet man wiederum  (mit Kennzeichnung der Pole durch + und -), ohne dass ein festes Prinzip erkennbar wäre, wann welches Zeichen bevorzugt wird. Schüler sollten beide Darstellungsarten kennenlernen.

Die Doppelbuchse  wird nur von `pst-circ` als eigenes Zeichen definiert (`\OpenDipol`). In `circuitikz` lässt sie sich mit Hilfe des Verbindungstyps `o-` durch zwei Einzelbuchsen realisieren (siehe Code zu Abb. 2). In `circuits.ee.IEC` fehlt die Unterstützung. Das Batteriezeichen  kennt `circuitikz` als `battery1` und `circuits.ee.IEC` als `battery`; `pst-circ` bietet nur die Darstellung  mit stärkerem Strich für den Minuspol (`\battery`), die in Schulbüchern unüblich ist.

Schaltelement	Aussehen	
	circuits.ee.IEC	schulma-physik
bulb		
capacitor		
amperemeter		
voltmeter		
messgeraet		
spannungsquelle		

Tab. 1: veränderte und ergänzte Schaltzeichen in schulma-physik

Bei Messgeräten für Spannung und Stromstärke existieren zwei konkurrierende Konventionen. Nach der einen steht $\text{---}\text{V}\text{---}$ für ein Voltmeter und $\text{---}\text{A}\text{---}$ für ein Amperemeter. Nach der anderen wird für beide $\text{---}\text{V}\text{---}$ (oder $\text{---}\text{A}\text{---}$) verwendet, wobei außen am Kreis das Formelzeichen der gemessenen Größe angegeben wird. Die Umsetzung der ersten Konvention wurde oben in den Beispielen für die verschiedenen Pakete demonstriert; für die zweite Konvention fehlt eine direkte Unterstützung leider in allen besprochenen Paketen.

Für Schalter sind verschiedene, leicht abweichende Schaltzeichen gebräuchlich. Ich fand $\text{---}\text{---}$, $\text{---}\text{---}$, $\text{---}\text{---}$ und $\text{---}\text{---}$. Die von unseren Paketen angebotenen Darstellungen sind alle brauchbar. Auch wenn evtl. keine völlige Übereinstimmung mit dem jeweiligen Schulbuch gegeben ist, dürfte der Wiedererkennungswert gegeben sein. Entsprechendes gilt für Umschalter, wobei `circuits.ee.IEC` keinen Umschalter anbietet.

Schaltbilder mit dem Paket `schulma-physik`

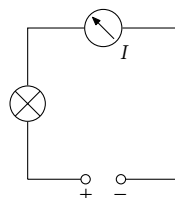
Im vorangegangenen Abschnitt ist deutlich geworden, dass keines der bisher vorgestellten Pakete die einfache Umsetzung aller Konventionen der Schulphysik in Schaltbildern erlaubt; gewisse Abstriche sind bei allen zu machen.

Das Paket `schulma-physik` aus dem Schulmathematik-Bündel nimmt Ergänzungen vor, die sich aus meinen Bedürfnissen als Physiklehrer ergeben, wobei natürlich persönliche gestalterische Vorstellungen eingeflossen sind. Die Grundlage bildet `circuits.ee.IEC`, vor allem aus dem Grund, dass ich mich hiermit zuerst beschäftigt habe.

schulma-physik lädt die TikZ-Bibliothek `circuits.ee.IEC`, definiert aber vier Schaltzeichen um und ergänzt zwei (Tabelle 1). Ein Blick auf Abbildung 3 zeigt, dass die Glühlampe in `circuits.ee.IEC` kleiner dargestellt wird als die ebenfalls kreisförmigen Symbole für Volt- und Amperemeter. Dies schien mir nicht ausgewogen, weshalb ich die Glühlampe vergrößert habe. Auch der Kondensator kam mir im Vergleich zu anderen Schaltzeichen zu unscheinbar vor. Bei Volt- und Amperemeter habe ich die Aufschrift auf serifenlos umgestellt, da man es meist so findet. Ergänzt wurden ein alternatives Schaltzeichen für Messgeräte und das in der Schule häufig vorkommende Zeichen für eine allgemeine Spannungsquelle.

Das neue Element `messgeraet` kann man mit der `circuits.ee.IEC`-Option `info` passend beschriften. Der Abstand der Beschriftung zum Schaltzeichen lässt sich durch die TikZ-Option `label distance` beeinflussen.

```
\begin{tikzpicture}[circuit ee IEC,font={\footnotesize}]
\draw (0,0) to[bulb] (0,2)
to[messgeraet={info=-45:{$I$}},label distance=-3pt]
(2,2)--(2,0) to[spannungsquelle]
node[xshift=-6.5pt,yshift=-6pt] {\strut $+$}
node[xshift=6.5pt,yshift=-6pt] {\strut $-$} (0,0);
\end{tikzpicture}
```



Wer `schulma-physik` mit `circuitikz` kombinieren will, sollte zwecks Vermeidung von Kompatibilitätsproblemen die gleichnamige Paketoption verwenden (verfügbar ab Version 1.4):

```
\usepackage[circuitikz]{schulma-physik}
```

Mit dieser Option wird `circuitikz` anstelle von `circuits.ee.IEC` geladen, und zwar mit den oben erklärten Optionen `resistor=european` und `inductor=american`; Änderungen am Schaltzeichenbestand werden in diesem Fall nicht vorgenommen. Unabhängig davon ist `schulma-physik` natürlich auch mit `pst-circ` kombinierbar.

Literatur

- [1] Herbert Voß: »Erstellen von Schaltbildern mit `pst-circ`«, *DTK*, 15.3 (2003), 33–49.

Musik-CD mit Cover produzieren

Ralf Mispelhorn

Vor einiger Zeit habe ich bereits einen Artikel über die Darstellung von Griffsymbolen für die Gitarre[3] und die Erzeugung von Song-Büchern[2] geschrieben. In diesem Artikel folgt ein Python-Skript, mit welchem eine Musik-CD erstellt wird und dabei auch das Cover mit L^AT_EX generiert wird.

Produktion der CD mit Cover

Einige CDs von mir sind entstanden, indem eine Word-Datei mit dem Cover und der Liste der Musik-Dateien ausgefüllt worden ist. Die Längen mussten ebenfalls von Hand eingetragen werden. Das Brennen der Titel auf CD erfolgte ebenfalls manuell. Mit Hilfe des Skripts ist es mir nun möglich, den ganzen Ablauf zu vereinfachen.

Liste der Musik-Dateien

Das Python-Skript `cover.py` erwartet als Argument ein Verzeichnis, in welchem sich die Musik-Dateien und das Cover-Bild `cover.png` befinden. Auf eventuelle Fehlerabfragen wird hier im Skript verzichtet, um den Umfang gering zu halten.

```
$ ./cover.py "Am blauen Meer"
```

Zuerst wird eine Liste der Musik-Dateien `*.wav` erstellt. Anschließend wird eine Referenz auf das Cover-Bild gebildet.

```
DIR = sys.argv[1]

# tracks
p = os.path.join (DIR, '*.wav' )
tracklist = [ fn for fn in glob.glob( p ) ]

# cover
cover = os.path.join ( DIR, 'cover.png' )
```

Umwandlung der Musik-Dateien

Die Musik-Dateien wurden zum größten Teil einfach mit dem Handy aufgenommen. Um sie später auf CD brennen zu können, müssen sie bestimmte Voraussetzungen

erfüllen (Stereo, 44100 Hz). Mit Hilfe des sox-Kommandos wurden von den Musik-Dateien Kopien mit dem erforderlichen Format in einem temporären Verzeichnis erstellt. Die Dateinamen wurden für den späteren Ablauf in einer Liste gespeichert. Anschließend wurde ebenfalls mit dem sox-Kommando die Audiolänge ermittelt. Die Länge wurde noch in das Format MM:SS gebracht.

```
def mm_ss ( sec ):
    m = int(sec / 60.)
    sec -= m * 60.
    s = int(sec )
    return '%d:%02d' % (m, s )
```

Der Titel ist der Dateiname ohne die Dateieindung. Beides wurde in einer Liste gespeichert. Zum Schluß wurde die Summe aller Audiolängen gebildet.

```
# prepare wav files
total = 0.
lst = []
tracklist2 = []

TMP = '_tmp'
if os.path.exists(TMP): shutil.rmtree ( TMP )
os.makedirs ( TMP )

for track,fn in enumerate ( tracklist ):
    bn = os.path.basename ( fn )
    dst = os.path.join (TMP, bn)

    tracklist2.append ( dst )

    # convert to stereo 44100 Hz
    cmd = f'sox -S "{fn}" "{dst}" channels 2 rate -L -s 44100 2>/dev/null >/dev/
↳null'
    os.system (cmd)

    # get audio length
    cmd = 'sox "%s" -n stat 2>&1 >/dev/null' % fn
    for line in os.popen ( cmd ):
        m = re.search ( '^Length \(\seconds\) : \s+(.*)', line )
        if m:
            sec = float(m.group(1))
            total += sec
            mmss = mm_ss ( sec )

    q = os.path.splitext ( bn )
```

```

print ( ' %2d: %s [%s]' % (track+1, q[0], mmss) )

t = ( q[0], mmss )
lst.append ( t )

mmss = mm_ss ( total )
print ( ' TOT: %s' % mmss)

```

Cover erzeugen

Für das Cover wurde das Paket `simplecd` von Dániel Majoros[1] verwendet, mit dem sich verschiedene Layouts für DVDs, CDs, VHS-Kassetten und ähnliches erzeugen lassen. Für die Musik-CD wurde das Makro `\covers` verwendet.

`\covers[backsheet]{cover}{spine}`

- `backsheet` (Rückseite): Alle Titel einschließlich der Länge der Titel
- `cover` (Vorderseite): Coverbild mit Titel der CD
- `spine` (Text für die Randseite): Titel der CD

```

# creating cover
fout = open ( 'cover.tex', 'w' )
sys.stdout = fout

print ( r'\documentclass{scrartcl}' )
print ( r'\usepackage[left=1cm,top=1cm,right=1cm,bottom=1cm]{geometry}' )
print ( r'\usepackage{graphics}' )
print ( r'\usepackage{simplecd}' )
print ( r'\usepackage{fontspec}' )
print ( r'\setmainfont{bitter}' )
print ( r'\begin{document}' )
print ( r'\covers' )
print ( r'[{ ' )
for n,t in enumerate(lst):
    i,ll = t
    print ( r'\hspace{1ex}%2d: %s\dotfill\s\hspace{1ex}~\\' % (n+1,i,ll) )

print ( r']}]' )
print ( r'{{\Huge \includegraphics[width=11cm,keepaspectratio]{%s} } \vspace{1
  ↳cm} \Huge %s }' % (cover,DIR) )
print ( r'{%s}' % DIR )
print ( r'\end{document}' )

```

```

sys.stdout = sys.__stdout__
fout.close()

cmd = 'lualatex cover.tex 2>/dev/null >/dev/null'
os.system ( cmd )

os.remove ( 'cover.aux' )
os.remove ( 'cover.log' )

```

Es entstand folgende L^AT_EX-Datei.

```

\documentclass{scrartcl}
\usepackage[left=1cm,top=1cm,right=1cm,bottom=1cm]{geometry}
\usepackage{graphics}
\usepackage{simplecd}
\usepackage{fontspec}
\setmainfont{bitter}
\begin{document}
\covers
[ {
\hspace{1ex} 1: Am blauen Meer\dotfill12:38\hspace{1ex}~\\
\hspace{1ex} 2: Am Träumen\dotfill13:17\hspace{1ex}~\\
\hspace{1ex} 3: Befreit\dotfill12:09\hspace{1ex}~\\
\hspace{1ex} 4: Café\dotfill12:25\hspace{1ex}~\\
\hspace{1ex} 5: Der Tag zu Ende\dotfill12:13\hspace{1ex}~\\
\hspace{1ex} 6: Ein neues Jahr\dotfill12:13\hspace{1ex}~\\
\hspace{1ex} 7: Genieß die Tage\dotfill12:02\hspace{1ex}~\\
\hspace{1ex} 8: Ich kenne ein Mädchen\dotfill12:09\hspace{1ex}~\\
\hspace{1ex} 9: Kaffee\dotfill12:42\hspace{1ex}~\\
\hspace{1ex}10: Langes Haar\dotfill11:58\hspace{1ex}~\\
\hspace{1ex}11: Lichter so bunt\dotfill12:29\hspace{1ex}~\\
\hspace{1ex}12: Nur für dich\dotfill13:29\hspace{1ex}~\\
\hspace{1ex}13: Regenbogen\dotfill12:18\hspace{1ex}~\\
\hspace{1ex}14: Schneeflocken\dotfill11:57\hspace{1ex}~\\
\hspace{1ex}15: Seifenschäum\dotfill12:11\hspace{1ex}~\\
\hspace{1ex}16: Vom Tag\dotfill13:40\hspace{1ex}~\\
\hspace{1ex}17: Wolken\dotfill12:21\hspace{1ex}~\\
} ]
{{\Huge \includegraphics[width=11cm,keepaspectratio]{Am blauen Meer/cover.png} }
\hspace{1cm} \vspace{1cm} \Huge Am blauen Meer }
{Am blauen Meer}
\end{document}

```

Nach dem Übersetzen ergab sich das folgende Cover-Dokument.



Am blauen Meer

1: Am blauen Meer.....	2:38	
2: Am Träumen	3:17	
3: Befreit.....	2:09	
4: Café.....	2:25	
5: Der Tag zu Ende	2:13	
6: Ein neues Jahr.....	2:13	
7: Genieß die Tage.....	2:02	
8: Ich kenne ein Mädchen	2:09	
9: Kaffee.....	2:42	
10: Langes Haar	1:58	
11: Lichter so bunt.....	2:29	
12: Nur für dich	3:29	
13: Regenbogen.....	2:18	
14: Schneeflocken	1:57	
15: Seifenschaum	2:11	
16: Vom Tag	3:40	
17: Wolken.....	2:21	

CD brennen

Zum Brennen wurde das wodim-Programm verwendet, welchem eine Liste aller Dateien übergeben werden muss. Hier mussten die einzelnen Titel in Quotes eingeschlossen werden, damit die Shell die Titel korrekt behandelt.

```
print ( 'CD creating ...' )
if not os.path.exists ( '/dev/cdrw' ):
```

```

print ( 'CD RW missing' )
sys.exit (1)

# build list of wav files
s = ''
for i in tracklist2:
    s += '"%s" ' % i

cmd = f'wodim -v speed=1 dev=/dev/cdrw -pad -audio {s} 2>&1'

n = 0
for line in os.popen ( cmd ):
    line = line.strip()
    print (line)
    m = re.search ( r'^(Track \d+: )Total bytes read', line )
    if m:
        x = os.path.basename(tracklist2[n])
        print ( ' ', m.group(1), os.path.splitext(x)[0] )
        n += 1

shutil.rmtree ( TMP )

```

Fazit

Nachdem die CD gebrannt war, musste nur noch das Cover gedruckt und ausgeschnitten werden. Durch dieses Skript war es nun möglich, Musik-CDs einfach zu erstellen. Erweitert wurde das Skript noch um eine Datei, welche die Reihenfolge der Titel beschreibt. Außerdem wurde noch das Songbook mit den Texten und den Gitarrenakkorden erstellt.

Literatur

- [1] Dániel Majoros: simplecd Package, <https://ctan.org/tex-archive/macros/latex/contrib/simplecd>.
- [2] Ralf Mispelhorn: »Erstellen eines Songbooks«, *DTK*, 34.2 (2022), 21–30.
- [3] — »Quintenzirkel mit Gitarren-Akkorden«, *DTK*, 34.2 (2022), 18–21.

Mehrsprachigkeit: Das L^AT_EX-Paket translator

Rolf Niepraschk

Mit Hilfe des L^AT_EX-Pakets translator (siehe [1]) lassen sich leicht Begriffe in die jeweils aktuell im L^AT_EX-Dokument eingestellte Sprache übersetzen. Man erreicht damit, dass die Übersetzungen bestimmter feststehender Begriffe zwangsläufig einheitlich geschehen. Besondere Bedeutung hat dies bei Mehrsprachigkeit unterstützenden Klassen und Paketen. So nutzt beispielsweise die Klasse beamer (siehe [2]) das Paket translator ausgiebig. Aber auch für eigene Projekte kann sich translator, wegen der einfachen Syntax, als nützlich erweisen.

Im Folgenden wird anhand eines einfachen Beispiels die Wirkungsweise gezeigt.

Das in der Abbildung dargestellte Tier wird folgendermaßen bezeichnet:

- Deutsch: Koboldmaki
- Englisch: Tarsier
- Niederländisch: Spookdiertje
- Italienisch: Tarsius
- Türkisch: Tarsier

Der relevante Code, der zu der gezeigten Ausgabe führt:

```
\usepackage[dutch,italian,ngerman]{translator}

\deftranslation[to = German]{Tarsier}{Koboldmaki}
\deftranslation[to = Dutch]{Tarsier}{Spookdiertje}
\deftranslation[to = Italian]{Tarsier}{Tarsius}
...
Das in der Abbildung dargestellte Tier wird folgendermaßen bezeichnet:
\begin{itemize}
  \item Deutsch:      \translate{Tarsier}
  \item Englisch:     \foreignlanguage{english}{\translate{Tarsier}}
  \item Niederländisch: \foreignlanguage{dutch}{\translate{Tarsier}}
  \item Italienisch:   \foreignlanguage{italian}{\translate{Tarsier}}
  \item Türkisch:     \foreignlanguage{turkish}{\translate{Tarsier}}
\end{itemize}
```

Wie man sieht, erfordert die englische Sprache nicht zwingend einen Übersetzungseintrag, da als Schlüsselwort (»Tarsier«) bereits der gesuchte englische Begriff genutzt und direkt verwendet wird. Im Falle des Türkischen wurde ebenfalls keine Übersetzung angegeben und daher wird auch hier ersatzweise das Schlüsselwort, also der englische Begriff, verwendet.



Abb. 1: Ein drolliges Äffchen!

Literatur

- [1] Joseph Wright: translator – Easy translation of strings in LaTeX, 2021, `macros/latex/contrib/translator/translator.pdf` (besucht am 24. 8. 2022).
- [2] Joseph Wright, Vedran Miletic, Till Tantau: The beamer class – User Guide for version 3.67, 2021, `macros/latex/contrib/beamer/doc/beameruserguide.pdf` (besucht am 24. 8. 2022).

PocketDiary – ein Modul für Terminkalender¹

Willi Egger

Das aktualisierte Con $\TeX$ t-Modul `pocketdiary` erzeugt personalisierte Mini-Terminplaner nach dem Vorbild von `PocketMod`, jetzt auch mit Mond- und Sonnendaten. Mit den Funktionen des Moduls lassen sich aber auch andere Arten von Kalendern erstellen, das zeigt der folgende Artikel.

Vorgeschichte

Die erste Version dieses Terminplaner-Moduls entstand im November 2010 für meinen Bruder Heinz. Damals bekam ich als Sekretär der NTG² die T $\TeX$ nische

¹ Dieser und der folgende Artikel werden im englischen Original im diesjährigen Journal der Context Group erscheinen; ins Deutsche übertragen und bearbeitet von Henning Hraban Ramm.

² Nederlandstalige T $\TeX$ Gebruikersgroep, <https://ntg.nl>

Komödie und las in Nr. 3/2010 den Artikel von Uwe Ziegenhagen [6] über seine L^AT_EX-Umsetzung von PocketMod³, einem Mini-Terminplaner mit acht Seiten auf einem Blatt A4-Papier. Da ich mit ConT_EXt arbeite(te) und schon verschiedene Ausschieß-Schemata für meine Buchprojekte eingerichtet hatte, entschloss ich mich, so etwas mit ConT_EXt und LuaT_EX umzusetzen. Das Projekt stellte ich auch bei der Frühjahrstagung von DANTE e.V. 2011 in Bremen vor.

Zehn Jahre später, beim ConT_EXt meeting 2021, zeigte Tomáš Hála seine Version eines Terminkalenders, den er mit ConT_EXt erzeugt hatte, als wegen der Covid-19-Pandemie alle Läden geschlossen hatten. In der anschließenden Diskussion kam die Frage auf, wie sich Mondphasen und die Zeiten von Sonnenauf- und -untergang darin integrieren ließen. Dabei erinnerte ich mich an meinen alten Terminkalender und wollte versuchen, diese Daten einzubauen.

Als ich meinen alten Code wiedergefunden hatte, musste ich feststellen, dass er mit ConT_EXt LMTX bzw. LuaMetaT_EX nicht mehr funktionierte – kein Wunder nach all den tiefgreifenden Veränderungen. Die nötigen Anpassungen hielten sich in Grenzen, die Berechnung der astronomischen Daten jedoch war eine Herausforderung.

Die Mondphasen waren noch kein großes Problem, da es relativ einfache Algorithmen dazu gibt. Astronomische Berechnungen werden allerdings schnell komplex, und bei vereinfachten Ansätzen muss man mit Schätzwerten leben können. Die verwendeten Formeln für die Mondphasen haben eine Toleranz von etwa einem Tag, aber das ist für unsere Zwecke gut genug.

Bei den Sonnendaten sah es anders aus. Im Internet fand ich mehrere Publikationen zu diesen Berechnungen, die Umsetzung der vorgestellten Algorithmen war allerdings nicht trivial. Obwohl ich annehme, dass die Darstellungen korrekt waren, reichte mein laienhaftes Verständnis nicht aus, um zu den gewünschten Ergebnissen zu kommen. Schließlich fand ich den Lua-Sourcecode für ein Display mit Sonnendaten, den ich für mein Modul anpassen konnte.

Umsetzung in Lua

Als ich im November 2010 anfang, dachte ich, es könnte nicht schwer sein, so einen Terminkalender umzusetzen. Das stimmt auch – solange man keine zu hohen Erwartungen hat.

Ich wollte eine leicht wartbare Produktionsdatei, für die einzelnen Wiederholungen sollten keine größeren Änderungen nötig sein. Das heißt aber, dass die Hintergrundprozesse alles aufgrund weniger Daten wie Jahr, Kalenderwoche bzw. Monat und Tag berechnen können müssen.

³ <https://www.pocketmod.com>

Mir ist schnell klargeworden, dass es im Kalenderjahr einfachere und schwierigere Abschnitte gibt. Einfacher sind Schaltjahre und alles außer Januar und Dezember. Anfang und Ende des Jahres sind besondere Momente: Sind z. B. die letzten Dezembertage in Woche 52 oder 53 und gehören die ersten Tage des Jahres zur letzten Woche des alten oder zur ersten des neuen?

Ich habe mich für die Berechnungen nach dem Europäischen Standard laut ISO 8601 entschieden. Danach ist die erste Woche eines Jahres die mit dem ersten Donnerstag. Außerdem ist festgelegt, dass die Woche am Montag beginnt – im Gegensatz zur Konvention von Lua, wo Tag 1 der Sonntag ist. Andererseits nimmt uns die os-Bibliothek von Lua einen großen Teil der Datumsberechnungen ab, also verlassen wir uns darauf so weit wie möglich.

Bei der Suche im Internet bekommt man eine Menge Einblicke in Datumsberechnungen. Die Arbeit von Sohael Babwani [1] ist eine interessante Quelle. Er beschreibt verschiedene Methoden zur Berechnung von Daten, die ich für das Modul verwendet habe. Da ich auch die wichtigsten christlichen Feiertage im Kalender haben wollte, war mir die Website von Ronald W. Mallen [2] bei der Astronomischen Gesellschaft von Südastralien eine große Hilfe. Er berechnet den Ostersonntag anhand von sechs Tabellen und bietet dafür auch ein BASIC-Programm an (umgesetzt von Greg Mallen), das ich nach Lua übertragen konnte.

Es ergaben sich weitere Fragen, z. B. wie man die Kalenderwoche eines Datums berechnet. Ich danke Ferry van Schaik [3] für einen Schnipsel Java, den ich wiederum nach Lua übersetzte.

Die Formel für die Berechnung der Tage im Mondzyklus habe ich bei SubsysTEMS [4] gefunden, einer amerikanischen Bildungseinrichtung. Im PocketDiary zeige ich die Nummer des Tages im Mondzyklus an, außer an Vollmond, Neumond sowie zu- und abnehmendem Halbmond; für diese Phasen habe ich Piktogramme in MetaFun erstellt. Die vereinfachte Berechnung berücksichtigt nicht alle Einzelheiten, daher ist sie nicht so genau wie andere, aber für die Mondphase ohne Angabe der Uhrzeit reicht es.

Auch für die Sonnendaten habe ich mich im Internet umgesehen. Wie bereits erwähnt, gibt es eine ganze Reihe von Artikeln dazu. Mir fehlt aber das Hintergrundwissen, um zu verstehen, was diese Formeln genau tun. Beim Ausprobieren verschiedener Berechnungen kam ich nie zu Ergebnissen, die ich in Stunden und Minuten konvertieren konnte. Nur dank der Arbeit von Alexander Yakushev [5] kann ich Aufgang und Untergang der Sonne berechnen; sein Code war bereits in Lua geschrieben, so dass ich ihn für die Anforderungen des PocketDiary-Moduls leicht anpassen konnte. Die Präzision kommt mir sehr hoch vor – verglichen mit

den Daten von Open Weather Map⁴ liegen die Abweichungen im Bereich von einer Minute.

Allgemeine Beschreibung

Ein PocketDiary ist ein persönlicher Terminplaner im Taschenformat, den man sich jeweils für eine Woche ausdruckt. Die acht Seiten sind so auf einem einseitig bedruckten A4-Blatt angeordnet, dass sie sich zu einem Heftchen falten lassen, bei dem nur bedruckte Seiten zu sehen sind.

Als Eingabe sollen jeweils nur Jahr und Kalenderwoche dienen, weitere Daten müssen nur einmalig eingerichtet werden.

Neben den Wochentagsseiten können auch Seiten mit Monats- und Jahreskalendern oder persönlichen Daten enthalten sein.

Dateien des Moduls

Das Modul kann aus dem »ConT_EXt garden«⁵ heruntergeladen werden.

Die Inhalte der Ordner `tex` und `doc` sortieren wir am besten in `texmf-modules` ein – oder in jeden anderen `texmf`-Baum, der nicht von der Distribution überschrieben wird. Danach rufen wir `mtxrun --generate` auf, damit ConT_EXt die Dateien finden kann.

Im Verzeichnis `tex/context/third/pocketdiary` liegen folgende Dateien:

- `PocketDiary.tex` Beispiel-Benutzerdatei
- `t-pocketdiary.tex` ist das eigentliche Modul und lädt einige Lua-Dateien.
- `t-pocketdiary.lua` enthält CLD⁶-Elemente.
- `t-calendar.lua` enthält die Kalender-Berechnungen.
- Verzeichnis `Moonphase`:
 - `t-moonphase.lua` enthält die Mondphasen-Berechnungen.
 - `moon-MP.tex` definiert die MetaFun-Piktogramme der Mondphasen.
- Verzeichnis `Solar`:
 - `t-solar.lua` enthält die Sonnen-Berechnungen.
 - `sun-MP.tex` definiert die MetaFun-Piktogramme für Sonnenauf- und untergang sowie Tageslicht-Stunden.

In `doc/context/third/pocketdiary` liegt die Dokumentation, die sich durch diesen Artikel erübrigt.

⁴ <https://openweathermap.org>

⁵ <https://modules.contextgarden.net/dl/>

⁶ ConT_EXt-Lua-Dokument

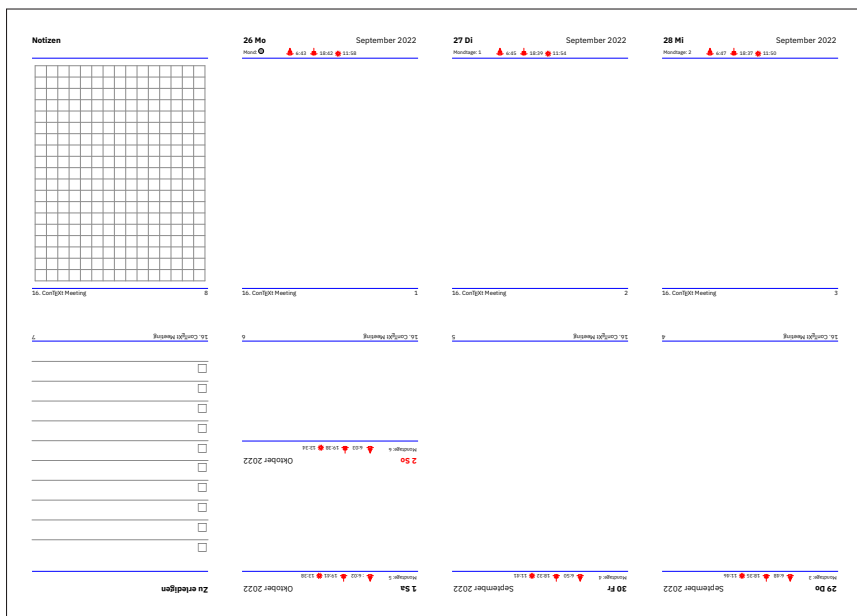


Abb. 1: Beispiel: Tageskalender mit Monats- und Jahreskalender

Einstellungen

Wir brauchen nur ein paar Zeilen Code für unseren persönlichen Terminkalender. Zuerst weisen wir ConT_EXt an, das Modul t-pocketdiary zu laden:

```
\usemodule[pocketdiary]
```

Wenn wir keine Grundschrift angeben, verwendet PocketDiary T_EX Gyre Pagella. Ansonsten können wir die Schrift hier einstellen:

```
\setupbodyfont[dejavu,ss,8pt]
```

Bei der Schriftgröße müssen wir darauf achten, dass die Seiten ziemlich klein sind. PocketDiary ist mehrsprachig, die Beschriftungen passen sich der eingestellten Hauptsprache an. Derzeit werden Deutsch, Englisch, Niederländisch, Italienisch, Spanisch und Französisch unterstützt.

```
\mainlanguage[de]
```

Das Seitenlayout verwendet Kopf- und Fußzeile sowie den Haupttextbereich. Die Kopfzeile enthält die Kalenderdaten in Abhängigkeit vom Seiteninhalt: Auf der

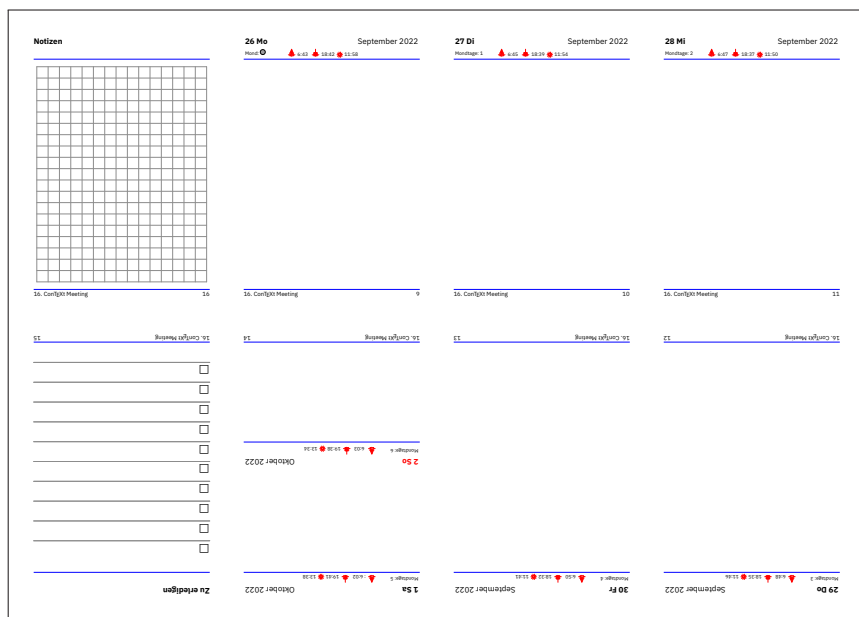


Abb. 2: Beispiel: Kalender für Woche, Monat und Folgemonat

Eintagesseite stehen der Tag im Monat, der Wochentagsname (abgekürzt), evtl. der Feiertagsname, die Kalenderwoche und das Jahr in der Kopfzeile. Darunter werden die Mondphase und die Sonnendaten (Aufgang, Untergang, Lichtstunden) ausgegeben. Wenn das Datum im Seiteninhalt steht, enthält die Kopfzeile Informationen wie Monat, Kalenderwoche und Jahr.

Alle Einstellungen sind über sechs Gruppen von Variablen zugänglich.

Die wichtigsten Kalendereinstellungen stehen in der Gruppe *PocketDiary*:

```
\setvariables
[PocketDiary]
[WeekDay=7,
 Week=28,
 Month=7,
 Year=2022,
 Nextyear=yes,
 Daybyday=no]
```

Variable	Wert	Beschreibung
WeekDay	Zahl	von 1 bis 7
Week	Zahl	von 1 bis 53
Month	Zahl	von 1 bis 12
Year	Zahl	Jahreszahl von 1900 bis 4099. Die untere Grenze hängt vom Betriebssystem ab, die obere von der Berechnung des Ostersonntags ([2]).
Nextyear	yes/no	Wenn »yes«, zeigt der Jahreskalender das nächste statt dem laufenden Jahr.
Daybyday	yes/no	Wenn »yes«, werden die Wochentage mit einer Seite pro Tag und das Wochenende als eine Seite mit Samstag und Sonntag ausgegeben. Damit werden sechs Seiten gefüllt, und wir können den Inhalt der Seiten 7 und 8 frei bestimmen.

Mit den Variablen der Gruppe *PocketDiaryLayout* können wir individuell festlegen, was auf den acht Seiten (Page1 bis Page8) erscheinen soll. Wir können problemlos das gleiche Muster mehrfach verwenden.

```
\setvariables
[PocketDiaryLayout]
[Page1=Lost-Return,
Page2=Weekcurrentplan,
Page3=Dayplan,
Page4=Monthcurrentplan,
Page5=Blank,
Page6=Contact,
Page7=Caro,
Page8=Lines]
```

Muster	Seiteninhalt
Dayplan	Tagesseite mit dem Datum, das durch die Variable »Week-Day« definiert ist.
Weekendplan	Das Wochenende der mit »Week« definierten Woche.
Weekcurrentplan	Wochenkalender für die Woche »Week«.
Weeknextplan	Wochenkalender für die Folgewoche (»Week« + 1).
Monthcurrentplan	Monatskalender für den Monat »Month«.
Monthnextplan	Monatskalender für den Folgemonat (»Month« + 1).
Yearcurrentplan	Jahreskalender für das Jahr »Year«.
Yearnextplan	Jahreskalender für das Folgejahr (»Year« + 1), wenn »Nextyear«=»yes«.
Lost-Returnto	Seite mit den Adressdaten aus der Variablengruppe <i>PocketDiaryAddress</i> .
Blank	Leere Seite (mit Kopf- und Fußzeile).
Todo	Leere Aufgabenliste.
Caro	Kariertes Papier.
Lines	Liniertes Papier.
Contact	Formular zum Eintragen von Kontaktdaten.

Die Variablengruppe *PocketDiaryAddress* enthält Daten für die Fußzeile und die Adress-Seite:

```
\setvariables[PocketDiaryAddress]
[Familyname=Egger,
 Forename=Willi,
 Street=Townstreet 3B,
 Zipcode=5000,
 City=New Beach,
 Country=TEX-world,
 Phone=+22 444 55 88 66,
 Mobile=+22 6 19 19 17 17,
 E-mail=info at pocketdiary.org,
 Web=www.pocketdiary.org]
```

Die Fußzeile enthält drei Felder, die sich mit den Variablen aus *PocketDiaryFooter* ändern lassen:

```
\setvariables[PocketDiaryFooter]
[Lefttext=PocketDiary,
 Centertext=\pagenumber,
 Righttext={\getvariable{PocketDiaryAddress}
 {Forename}},~{\currentdate[year]}]
```

Die Daten für die Sonnenberechnungen müssen wir in *PocketDiaryGeoPosition* eintragen; sie hängen vom Ort ab, für den die Daten gelten sollen:

```
\setvariables[PocketDiaryGeoPosition]
[lat=52.467860,    % Breite
 lon=16.981240,    % Länge
 timezoneoffset=1, % Zeitversatz
 continent=EU]      % EU, US, leer
```

Variable	Bedeutung
lat	Breitengrad
lon	Längengrad
timeoffset	Zeitversatz von UTC in Stunden
continent	Sommerzeit-Regeln: EU oder US

Die Sommerzeit (daylight saving time, DST) beginnt je nach Land an verschiedenen Tagen, wenn überhaupt.

- In **Europa** beginnt die Sommerzeit am letzten Sonntag im März und endet am letzten Sonntag im Oktober.
- In den **USA** beginnt die Sommerzeit am zweiten Sonntag im März und endet am ersten Sonntag im November (laut Energy Policy Act von 2005) – außer in Arizona, wo es keine gibt.

Welche Regeln angewendet werden, hängt von der Variable *continent* ab; mit den Werten EU oder US werden Beginn und Ende der Sommerzeit entsprechend berechnet; ist sie leer, wird keine Sommerzeit berechnet.

Damit die Zeiten für Sonnenauf- und -untergang stimmen, müssen wir Längen- und Breitengrad unseres Bezugsortes in *lon* und *lat* eintragen. Außerdem muss der Versatz zur Universalzeit (UTC) in *timeoffset* stimmen.

Die Sonnendaten werden auf jeder Tages- und Wochenendseite ausgegeben. Auf dem Wochenkalender werden die Daten nur am Dienstag angezeigt.

Kopf- und Fußzeile sind durch eine Linie vom Textbereich abgegrenzt; ihre Farbe »Separatorline« ist standardmäßig blau. Rasterlinien »Gridline« sind normalerweise hellgrau. Wir können beides in der Variablengruppe *PocketDiaryColors* verändern:

```
\setvariables[PocketDiaryColors][
  Separatorline=blue,
  Gridline={s=.55},]
```

Mit den letzten Zeilen lassen wir das Ganze ausgeben:

```
\starttext
\setuparranging[1*8]
```

```
\getbuffer[PocketDiary]
\stoptext
```

Die Angabe zum Ausschieß-Schema `\setuparranging[1*8]` könnte auch in `t-pocketdiary.tex` stehen, es erleichtert aber die Fehlersuche, wenn wir die Zeile hier verändern können.

Seitenmuster

Kalenderseiten



Abb. 3: Dayplan

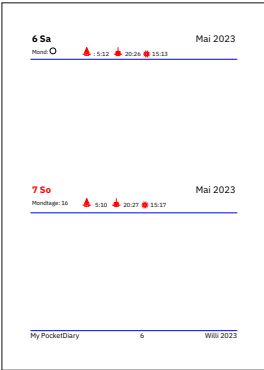


Abb. 4: Weekendplan

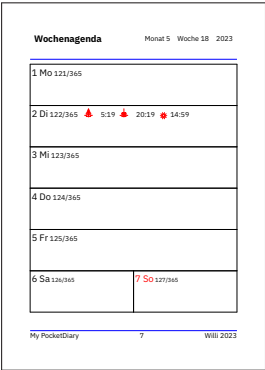


Abb. 5: Week

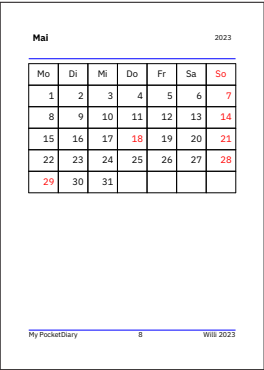


Abb. 6: Month

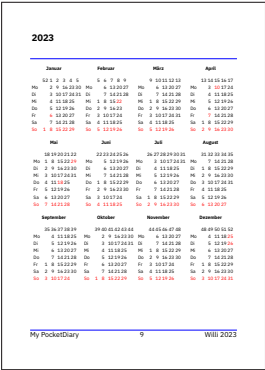


Abb. 7: Year

Notizen usw.

Abb. 8: Blank

Abb. 9: Caro

Abb. 10: Lines

Abb. 11: Todo

Abb. 12: Contact

Abb. 13: Lost-Returnto

Faltanleitung

Das ausgedruckte Blatt mit den acht Seiten wird so gefaltet, dass sich ein Heftchen ergibt, in dem die unbedruckten Seiten nicht zu sehen sind.

Wir legen das Blatt mit der bedruckten Seite nach unten vor uns und machen zwei Bergfalze an den geraden Linien in Bild 14. Dann falten wir es wieder auf, wenden es (bedruckte Seite nach oben) und machen zwei Talfalze an den gestrichelten Linien bis zur Mitte, und falten es wieder auf – nun sind die acht Seiten erkennbar.

Mit einem scharfen Messer und einem Lineal (oder mit einer Schere) schneiden wir das Papier zwischen den Punkten A und B auf (siehe Bild 14).

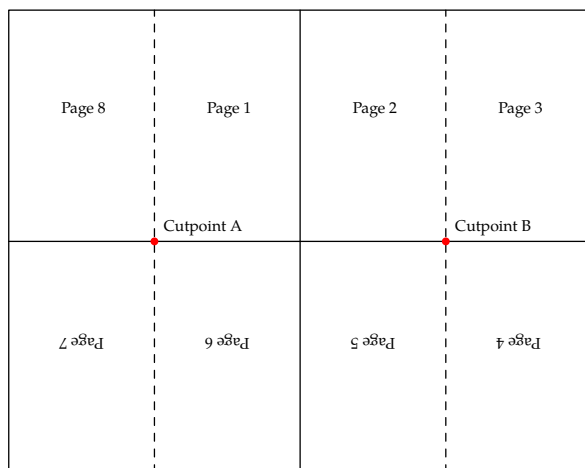


Abb. 14: Faltschema 1

Jetzt können wir das Heft falten: Zuerst falten wir das Papier wieder längs. Dann halten wir das gefaltete Papier mit dem Bergfalz nach oben und schieben von beiden Seiten zur Mitte, um eine Form zu bekommen wie in Bild 15. Dann falten wir die obere Doppelseite in Richtung B, die untere in Richtung C und schließlich die linke Doppelseite in Richtung D.

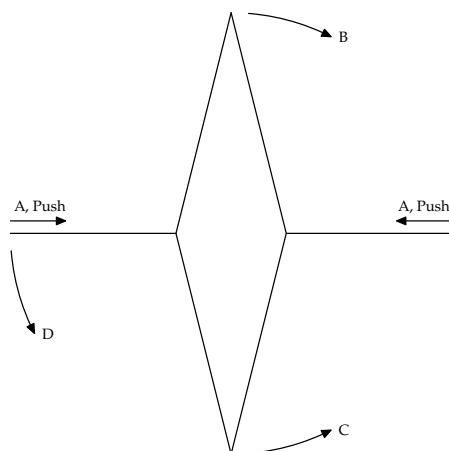


Abb. 15: Faltschema 2

Bevor wir den Bund falzen, lohnt es sich, das Heft auf den Tisch zu legen und die Falze nachzujustieren, soweit nötig. Zum Schluss glätten wir den Bundfalz, am besten mit einem Falzbein. Zur Perfektionierung können wir das Heft nun mit Nadel und Faden rückstichheften.

Danksagung

Ich möchte Hans Hagen für ConT_EXt mit LuaMetaT_EX danken und für all seine Geduld, während er mir half, die Probleme zu lösen, über die ich unterwegs stolperte. Außerdem danke ich Wolfgang Schuster für seine Unterstützung bei der mehrsprachigen Umsetzung der ersten Version des Moduls und schließlich Hraban Ramm, der mich zu Detailverbesserungen gedrängt hat.

Literatur

- [1] Sohael Bibwani: An extended Approach to the Julian and Gregorian Calendar, Babwani's Congruence and Babwani's Algorithm, 2010, <http://www.babwani-congruence.blogspot.com/>.
- [2] Ronald W. Mallen: Easter Dating Method, Astronomical Society of Southern Australia, 2002, <https://www.assa.org.au/edm.html>.
- [3] Ferry van Schaik: How can I calculate the week number of the current date?, Internet Related Technologies, 2001, <https://www.irt.org/script/914.htm>.
- [4] Subsystems: Calculate the Moon Phase, 2017, <https://www.subsystems.us>.
- [5] Alexander Yakushev: Module for calculating sunrise/sunset times for a given location based on algorithm by United States Naval Observatory, Washington, Version revision 3, 2022, <https://gist.github.com/alexander-yakushev/88531e23a89a0f2acb1>.
- [6] Uwe Ziegenhagen: »PocketMods mit L^AT_EX erstellen«, *DTK*, 22.3 (2010), 27–32.

Andere Kalender mit dem PocketDiary-Modul

Willi Egger

Da wir nun die ganze Kalender-Maschinerie für die PocketDiaries eingerichtet haben, sollten wir sie doch auch für andere Kalender nutzen können? Bitte sehr – hier kommen ein Fotokalender, ein Essensplan und ein Jahreskalender mit Sonne und Mond (letztere als Heft).

Einführung

Wenn wir uns die Möglichkeiten ansehen, die das PocketDiary-Modul bietet, sollten sich damit allerlei andere Kalender generieren lassen. Wir müssen nur das Modul laden und können mit ein wenig Fantasie all die Funktionen sowie die grundlegenden Formate für Kalender-Tabellen zu anderen Zwecken verwenden. Der Code der Beispiele ist ebenso wie die Module im »ConTeXt garden«¹ verfügbar.

Fotokalender

Zunächst basteln wir uns einen Foto-Monatskalender zum Umblättern. Der obere Teil jeder Seite soll ein Foto sowie den Monatsnamen enthalten, der untere Teil eine Monatstabelle mit den Wochentagen in der ersten Spalte. Die Sonntage sollen in rot ausgegeben werden.

Zunächst laden wir das pocketdiary-Modul und stellen die Hauptsprache, das Seitenformat sowie die Grundschrift ein:

```
\usemodule[pocketdiary]
\mainlanguage[de]
\setuppapersize[A4]
\setupbodyfont[dejavu,ss,12pt]
```

Wir brauchen keine Seitenzahlen und passen das Layout auf die Größe des Kalenders an:

```
\setuppagenumbering[location=]
\setuplayout[
  topspace=20mm,
  backspace=8.6mm,
  header=0pt,
  footer=0pt,
  height=middle,
  width=middle]
```

Für Titelseite und Monate brauchen wir 13 Fotos. Wir teilen ConTeXt mit, wo wir sie abgelegt haben:

```
\setupexternalfigures[directory={../kalenderfotos}]
```

Nachdem die Bilder später mit Hilfe einer Programmierschleife platziert werden, vergeben wir für die Bilder symbolische Namen bzw. Nummern:

¹ <https://modules.contextgarden.net/dl/>

```
\useexternalfigure [1] [Birkenteller]
% ...
\useexternalfigure [13] [Kantele]
```

Von all den Variablen aus dem PocketDiary brauchen wir nur das Jahr definieren:

```
\setvariables[PocketDiary][Year=2023]
```

Dieses Beispiel kann mit mehreren Sprachen umgehen, für die wir hier die Titelbezeichnung definieren:

```
\setuplabeltext[en][fotocal=Photo Calendar]
\setuplabeltext[nl][fotocal=Fotokalender]
\setuplabeltext[de][fotocal=Fotokalender]
\setuplabeltext[fr][fotocal=Calendrier photo]
\setuplabeltext[it][fotocal=Calendario fotografico]
```

Jetzt können wir den Kalender in Lua erzeugen:

```
\startluacode
  local report = logs.reporter("Fotokalender")

  function thirddata.calendar.Photo_calendar_page(y)

    report("Working in function: calendar.Photo_calendar_page")

    local year = y
    for i = 1,12 do

      local monthname = string.lower(os.date("%B",
        os.time{year=year,month=i,day=1}))

      context.startplacefigure({number="",title=""})
        context.externalfigure({i}, {width = "0.8\\textwidth" })
      context.stopplacefigure()
      context.strut()
      context("\\vfill")
      context.hfil()
      context("\\bfa")
      context.labeltext(monthname)
      context("~")
      context(year)
      context.hfil()
      context("\\tf")
      context.blank()
```

```

        thirddata.diary.monthtableH(i,year)

        context.page()
    end
end
\stopluacode

```

Das Tabellenlayout müssen wir ein bisschen anpassen, dazu verpacken wir die Einstellungen in ein setup:

```

\startsetups table:month
\setupTABLE[c][each]
  [width=\dimexpr\textwidth/7\relax, frame=on, style=\tfa]
\setupTABLE[r][each]
  [height=1.8\lineheight, align={center, lohi}]
\setupTABLE[c][7]
  [framecolor=black, foregroundcolor=red]
\stopsetups

```

Für den Aufruf der Lua-Funktion definieren wir uns ein Makro:

```

\define[1]\Photocalendarpages
  {\ctxlua{thirddata.calendar.Photo_calendar_page(#1)}}

```

Die Titelseite erzeugen wir als standardmakeup:

```

\starttext

\startstandardmakeup[page=yes, doublesided=yes]
  \startplacefigure[
    location=middle,
    title=,
    number=]
    {\externalfigure[13][width=\paperwidth]}
  \stopplacefigure
  \midaligned{\bfc \labeltext{fotocal}~\getvariable{PocketDiary}{Year}}
\stopstandardmakeup

\Photocalendarpages{\getvariable{PocketDiary}{Year}}

```

Das Ergebnis sieht so aus:



Abb. 1: Titelseite

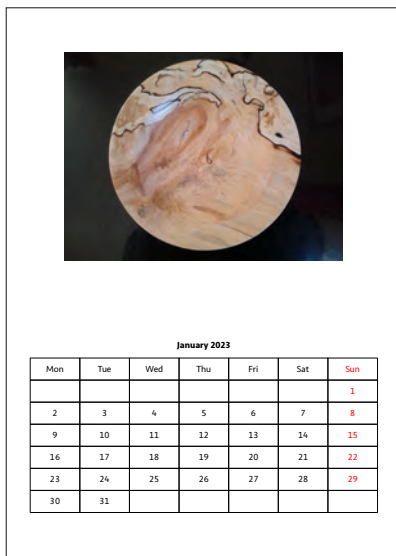


Abb. 2: Januar 2023

Menüplan

Angenommen, wir möchten jeden Tag aufschreiben, was wir gegessen haben, einfach weil uns das interessiert oder um eine bestimmte Diät einzuhalten. Dazu erzeugen wir uns ein Heft mit einer Woche pro Doppelseite. Die erste (ungerade) Seite füllen wir mit einem Bild.

In der Kopfzeile geben wir den Monat und die Kalenderwoche aus, in der Fußzeile den Dateinamen, Seitenzahl und Jahr. Jeden Tag können wir eintragen, was wir gekocht haben und kommentieren, wie es geschmeckt hat.

Ganz ähnlich ließe sich das für Hausaufgaben oder ein Arbeitsprotokoll verwenden. Zuerst laden wir die Module `pocketdiary` und `collatingmarks` und definieren die Hauptsprache:

```
\usemodule[pocketdiary]
\usemodule[collatingmarks][Collatingmarks=yes]
\mainlanguage[de]
```

PocketDiary stellt die Grundschrift auf Pagella ein, wir möchten aber lieber eine andere Schrift:

```
\setupbodyfont[ibmplex,rm,8pt]
```

Die Paginierung soll darauf Rücksicht nehmen, dass wir Doppelseiten verwenden:

```
\setuppagenumbering[alternative=doublesided]
```

Das Ausschieß-Schema ist »doppelseitig, je vier Seiten auf Schön- und Widerdruck auf zwei Bogen«, somit hat ein Heft 16 Seiten:

```
\setuparranging[2*4*2]
```

So könnte unser Seitenlayout aussehen:

```
\setuplayout[
  topspace=.6cm,
  backspace=.6cm,
  header=2\bodyfontsize,
  headerdistance=.5\bodyfontsize,
  footer=1.2\bodyfontsize,
  footerdistance=.5\bodyfontsize,
  margin=0pt,
  height=middle,
  width=middle,
  location=middle]
```

Von den PocketDiary-Variablen brauchen wir wieder nur das Jahr:

```
\setvariables[PocketDiary][Year=2023]
```

Die Fußzeile können wir individuell gestalten. Hier verwenden wir `\labeltext`, um die Mehrsprachigkeit zu erleichtern:

```
\setvariables[PocketDiaryFooter][
  Lefttext=\labeltext{menu}-\labeltext{calendar},
  Centertext=\pagenumber,
  Righttext=\getvariable{PocketDiary}{Year}]
```

PocketDiary verwendet eine blaue Linie zur Abtrennung von Kopf- und Fußzeile; das könnten wir ändern:

```
\setvariables[PocketDiaryColors][
  Separatorline=blue,
  Gridline={s=.75}]
```

Die Einstellungen für die Wochentabelle verpacken wir in ein `setup`:

```

\startsetups table:week
  \setupTABLE[split=yes]
  \setupTABLE[c][1][width=.35\textwidth,frame=off]
  \setupTABLE[c][2][width=.65\textwidth,frame=off]
  \setupTABLE[r][height=2\lineheight,align=lohi]
  \setupTABLE[r][2,3,5,6,8,9,11,12,14,15,17,18,20,21][bottomframe=on]
\stopsetups

```

Die automatisch eingesetzten Texte definieren wir in mehreren Sprachen:

```

\setuplabeltext[en][menu=Menu]
\setuplabeltext[en][comment=Comment]

\setuplabeltext[de][menu=Menü]
\setuplabeltext[de][comment=Kommentar]

\setuplabeltext[fr][menu=Menu]
\setuplabeltext[fr][comment=Commentaire]

% ...

```

Die Magie geschieht weitgehend in Lua. Dieser Codeblock wird für die Wochen 52-mal aufgerufen:

```

\startluacode
local report = logs.reporter("Menükalender")

function thirddata.calendar.menuweektable(mondaytimestamp,year,w)
  local s = mondaytimestamp
  local week = w
  local d = nil
  local n = nil
  local f = nil
  local m = nil
  local day = s + 1 * 24 * 60 * 60

  report("Working in function: thirddata.calendar.menuweektable")
  --report("Day-beforeloop: %s", day)

  local monthname = string.lower(os.date("%B",day))

  thirddata.calendar.myheadertext(monthname,week)

  --report("Monthname: %s", monthname)

```

```

context.bTABLE({setups="table:week"})
  for i = 1,7 do
    day = s + i * 24 * 60 * 60

    --report("Day-inloop: %s", day)

    d = tonumber(os.date("%d",day))
    n = string.lower(os.date("%a",day))
    m = tonumber(os.date("%m",day))
    f = thirddata.calendar.checkchristianfeast(d,m,year)
    if not f then
      f = " "
    end

    --report("Day %s", n)

    context.bTR()
    context.bTD({nx=2,style="bfa"})
    if i == 7 then
      context.color({"red"},function() context(d) end) -- Hans
      context("~")
      context.color({"red"},function() context.labeltext(n) end)
      context("~")
    else
      context(d)
      context("~")
      context.labeltext(n)
    end
    context.style({"tfx"})
    context.labeltext(f)
    context.eTD()
  context.eTR()
  context.bTR()
    context.bTD()
      context.labeltext("menu")
    context.eTD()
    context.bTD()
    context.eTD()
  context.eTR()
  context.bTR()
    context.bTD()
      context.labeltext("comment")
    context.eTD()

```

```

        context.bTD()
        context.eTD()
        context.eTR()
    end
    context.page({"yes"})
    context.eTABLE()
    context.page({"yes"})
end

function thirddata.calendar.myheadertext(monthname, week)
    context.setupheadertexts{
        function()
            context("\bgroup\bfa")
            context.labeltext(monthname)
            context("\egroup")
            context("\hfill")
            context.labeltext("week")
            context(": ")
            context(week)
            return true
        end}
end

function thirddata.calendar.week(weeknumber, year)
    local w = weeknumber
    local mondaytimestamp = thirddata.calendar.weekcalendar(w, year)

    report("Working in function: thirddata.calendar.week")
    --report("Mondaytimestamp %s", mondaytimestamp)

    thirddata.calendar.menuweektable(mondaytimestamp, year, w)
end

\stopluacode

```

Um die Lua-Funktion aufzurufen, brauchen wir ein Makro mit Kalenderwoche und Jahr als Parametern:

```
\define[2]\MenuWeektable{\ctxlua{thirddata.calendar.week(#1,#2)}}
```

Das Modul `collatingmarks` erzeugt Kollationierungsmarken für die Reihenfolge der ausgeschossenen Hefte. Dazu müssen wir ein paar Parameter festlegen.

Jedes Blatt Papier enthält acht Seiten (je vier auf zwei Seiten), wir verwenden zwei Blätter. Wir brauchen keine horizontale Verschiebung der Kollationierungsmarken.

```
\setupMPvariables[
  pages=8,    % Seiten pro Blatt
  sheets=2,   % Blätter pro Heft
  horpageshift=0mm]
```

Jetzt haben wir alles vorbereitet und können das Heft setzen. Auch hier verwenden wir standardmakeup für die Titelseite:

```
\starttext
% Titelseite
\startstandardmakeup[page=yes,doublesided=yes]
\startalignment [middle]
  \godown[.3\textheight]
  {\bfc \labeltext{menu}-\labeltext{calendar}}
  \vfill
  {\tfa \getvariable{PocketDiary}{Year}}
\stopalignment
\vfil
\stopstandardmakeup

\setupheadertexts[[]]
\strut\vfil
\placefigure
  [middle,none]
  [reference]
  {none}
  {\externalfigure[Erdbeertoertli][width=0.75\textwidth]}
\vfill

\page

\setups{Footertext}
\dorecurse{52}
  {\setvariables
    [PocketDiary]
    [Week=\recurselevel]
    \MenuWeektable{\recurselevel}{\getvariable{PocketDiary}{Year}}\page}
\page
\setupheadertexts[\bf \labeltext{notes}]
\dorecurse{3}
  {\strut\page}
```

Und so sieht das Ergebnis aus:

January Week: 1

3 Mon

Menue _____

Comment _____

4 Tue

Menue _____

Comment _____

5 Wed

Menue _____

Comment _____

6 Thu Epiphany

Menue _____

Comment _____

Menue-Calendar 2 2022

Abb. 3: Gerade Seite

January Week: 1

7 Fri

Menue _____

Comment _____

8 Sat

Menue _____

Comment _____

9 Sun

Menue _____

Comment _____

Menue-Calendar 3 2022

Abb. 4: Ungerade Seite

Kalender mit Sonnen- und Monddaten

Für meine Naturbeobachtungen hätte ich gerne einen kleinen Kalender mit den Zeiten des Sonnenauf- und -untergangs sowie den Mondphasen. PocketDiary bringt die nötigen Funktionen mit.

Zuerst laden wir wieder die Module und setzen die Hauptsprache:

```
\usemodule[pocketdiary]
\usemodule[collatingmarks][Collatingmarks=yes]
\mainlanguage[de]
```

Diesmal brauchen wir das Jahr und die Ortsangaben in den Feldern »Place« und »Country«:

```
\setvariables[PocketDiary][
  Year=2023,
  Place=Sambeek,
  Country=NL]
```

In der Fußzeile geben wir den Dateinamen und die Seitenzahl an:

```
\setvariables[PocketDiaryFooter][
  Lefttext={\jobname .pdf},
  Centertext=\pagenumber,
  Righttext=]
```

Wir können auch die Trennlinie für Kopf- und Fußzeile anpassen:

```
\setvariables[PocketDiaryColors][
  Separatorline=blue,
  Gridline={s=.75}]
```

Am wichtigsten sind die Ortsangaben, die für die eigentliche Berechnung benötigt werden:

```
\setvariables[PocketDiaryGeoPosition][
  lat=51.63682, % Sambeek
  lon=5.96655, % Sambeek
  timezoneoffset=1,
  continent=EU] % Sommerzeit
```

Wir stellen die Standardschriftart und das Seitenlayout ein:

```
\setupbodyfont[ibmplex,ss,10pt]
\setuppapersize[A6][A4]
\setuplayout[
  topspace=.6cm,
  backspace=.6cm,
  header=2\bodyfontsize,
  headerdistance=.5\bodyfontsize,
  footer=1.2\bodyfontsize,
  footerdistance=.5\bodyfontsize,
  margin=0pt,
  height=middle,
  width=88mm,
  location=middle]
\setuppagenumbering[alternative=doublesided]
```

Das Buch wird in zwei Hefte aus je zwei Blättern ausgeschossen; jeder Bogen enthält je vier Seiten auf Vorder- und Rückseite. Dafür müssen auch die Kollationsmarken eingerichtet werden:

```
\setuparranging[2*4*2]
\setupMPvariables[ % collatingmarks
  pages=8, % Seiten pro Bogen
  sheets=2, % Bogen pro Heft]
```

```
horpageshift=0mm]
```

Das meiste passiert wieder im Lua-Code:

```
\startluacode
local report = logs.reporter("Sundata calendar")

function thirddata.diary.year_of_monthtables(
  year,lat,lon,timeoffset,continent)

  report("Working in function: diary.year_of_monthtables")

  -- Determination of the DST start and stop date
  local dst_start,dst_stop = thirddata.diary.DST(year,continent)
  --report("DST start: %s, DST stop: %s",dst_start,dst_stop)

  for i= 1,1 do
    local monthname = string.lower(os.date
      ("%B",os.time{year=year,month=i,day=1}))
    context.setupheadertexts(
      {
        function()
          context("\bf")
          context.labeltext(monthname)
          return true
        end
      },
      {
        function()
          context(year)
          return true
        end
      },
      {
        function()
          context("\bf")
          context.labeltext(monthname)
          return true
        end
      },
      {
        function()
          context(year)
          return true
        end
      }
    )
  end
end
```

```

        end
    })

    thirddata.diary.monthblock(i,year,lat,lon,
        timeoffset,dst_start,dst_stop)
end
end

function thirddata.diary.monthblock(
    month,year,lat,lon,timeoffset,dst_start,dst_stop)

    report("Working in function diary.monthblock")

    -- returns number of weeks in a given month:
    local c = thirddata.calendar.month(month,year)
    -- returns the week number of the 1st day of a month:
    local wkf = thirddata.calendar.weeknumber(1,month,year)
    local nofdays = 1

    if month == 2 and thirddata.calendar.isleapyear(year) ~= 365 then
        nofdays = thirddata.calendar.nofdays_month(month,year) + 1
    else
        nofdays = thirddata.calendar.nofdays_month(month,year)
    end

    --report("Number of days in month: %s", nofdays)
    -- returns the week number of the last day of a month.

    local wk1 = thirddata.calendar.weeknumber(nofdays,month,year)
    local monthname = thirddata.calendar.select_nameofmonth(month,year)

    context.bTABLE({setups="table:monthblock,split=yes"})

    for i = 1,nofdays do
        local ordinalday = thirddata.calendar.ordinalday(i,month,year)
        local weeknumber = thirddata.calendar.weeknumber(i,month,year)
        local holiday = thirddata.calendar.checkchristianfeast(i,month,year)
        local moondata = thirddata.moonphase.lunardays(year,month,i)
        local sunrise,sunset,lighthours =
            thirddata.srss.sundata(
                i,month,year,lat,lon,timeoffset,dst_start,dst_stop)

        context.bTR()
        context.bTD()
    end
end

```

```

local dayname = string.lower(os.date
    ("%a",os.time{day=i,month=month,year=year}))

--report("Dayname: %s",dayname)
context.starttabulate{"|lw(0.1\\textwidth)|lw(0.06\\textwidth)|l|"}
context.NC()
    if dayname == "sun"
        or holiday ~= "" and holiday ~= "asw" then
            local dnu = i
            local dna = dayname
            context.color({"red"},
                function() context(dnu) end) --day number
            context("~")
            context.color({"red"},
                function() context.labeltext(dna) end) --dayname
        else
            context(i) --day number
            context("~")
            context.labeltext(dayname) --dayname
        end
context.NC()
context.switchtobodyfont{"6pt"}
context.labeltext("week")
context(": ")
context(weeknumber)
context.NC()
context.switchtobodyfont{"6pt"}
context.labeltext("day")
context(": ")
context(ordinalday)
context.NC()
context.NR()
context.stoptabulate()
context.eTD()
context.bTD()
context.starttabulate{"|lw(0.2\\textwidth)|l|l|l|l|"}
context.NC()
    if holiday == "" then
        context.strut()
        context.NC()
        context.NC()
        context.NC()
        context.NC()
    end

```

```

else
  context.labeltext(holiday)
  context.NC()
  context.NC()
  context.NC()
  context.NC()
end
context.NR()
context.NC()
  if tonumber(moondata) then
    context.labeltext("moondays")
  else
    context.labeltext("moon")
  end
  context("::~")
  context(moondata)
context.NC()
  --context.labeltext("sunrise")
  context("\SunA[background=Sunrise]{}")
  context("\quad ::")
  context(sunrise)
context.NC()
  --context.labeltext("sunset")
  context("\SunA[background=Sunset]{}")
  context("\quad ::")
  context(sunset)
context.NC()
  --context.labeltext("lighthours")
  context("\SunB[background=Light]{}")
  context("::~")
  context(lighthours)
  context.NC()
  context.NR()
  context.stoptabulate()
  context.eTD()
  context.eTR()
end
context.eTABLE()
context.page()
end
\stoptuacode

```

Das Tabellenlayout ist in eine setup-Umgebung verpackt:

```
\startsetups[table:monthblock]
  \setupTABLE[split=yes]
  \setupTABLE[each][each][frame=on]
  \setupTABLE[r][each][offset=0pt,loffset=3pt,roffset=3pt,align=lohi]
  \setupTABLE[c][1][width=0.35\textwidth,align=lohi]
  \setupTABLE[c][2][style={\switchtobodyfont[6pt]},width=0.65\textwidth] \
↪stopsetups
```

Um die Lua-Funktion aufzurufen, deklarieren wir ein Makro mit fünf Parametern:

```
\define[5]\Yearofmonthtables % Year,lat,lon,timeoffset,continent
  {\ctxlua{thirddata.diary.year_of_monthtables(#1,#2,#3,#4,#5)}}
```

Das Makro mit den gesammelten Informationen stecken wir in einen Puffer:

```
\startbuffer[Yearofmonthtables]
  \Yearofmonthtables
    {\getvariable{PocketDiary}{Year}}
    {\getvariable{PocketDiaryGeoPosition}{lat}}
    {\getvariable{PocketDiaryGeoPosition}{lon}}
    {\getvariable{PocketDiaryGeoPosition}{timeoffset}}
    {"\getvariable{PocketDiaryGeoPosition}{continent}"}
\stopbuffer
```

Um das Dokument zu internationalisieren, so dass wir ggf. nur die Hauptsprache umstellen müssen, deklarieren wir einige Label:

```
\setuplabeltext[en][title=Title]
\setuplabeltext[de][title=Titel]
% ...
\setuplabeltext[en][author=Author]
\setuplabeltext[de][author=Autor]
% ...
\setuplabeltext[en][typography=Typography]
\setuplabeltext[de][typography=Typographie]
% ...
\setuplabeltext[en][compilation=Compilation date]
\setuplabeltext[de][compilation=Kompilationsdatum]
% ...
```

Damit haben wir alles beisammen, um das Dokument zu setzen:

```
\starttext

\startstandardmakeup[page=yes,doublesided=no]
```

```

\strut
\startalignment [middle]
  \godown[.3\textheight]
  \startmode[**en]
    {\bfc\labeltext{sunstar}- and
      \labeltext{moon}-\labeltext{calendar}}
  \stopmode
  \startmode[**de]
    {\bfc\labeltext{sunstar}n- und
      \labeltext{moon}-\labeltext{calendar} }
  \stopmode
  % ...
  \blank
  {\bfb\getvariable{PocketDiary}{Place}}
  \vfill
  {\tfa\getvariable{PocketDiary}{Year}}
\stopalignment
\vfil
\stopstandardmakeup

\startstandardmakeup[page=yes,doublesided=no]
\starttabulate[|l|p|]
  \NC \labeltext{title}
  \EQ
  \startmode[**en]
    {\labeltext{sunstar}- and
      \labeltext{moon}-\labeltext{calendar}}
  \stopmode
  \startmode[**de]
    {\labeltext{sunstar}n- und
      \labeltext{moon}-\labeltext{calendar} }
  \stopmode
  % ...
  \NC\NR
  \NC \labeltext{author}
  \EQ \getvariable{PocketDiaryAddress}{Forename}~
    \getvariable{PocketDiaryAddress}{Familyname} \NC\NR
  \NC \labeltext{typography} \EQ \CONTEXT\ with \LUAMETATEX \NC\NR
  \NC \labeltext[compilation] \EQ \currentdate \NC\NR
\stoptabulate
\vfil
\startmode[**en]
  This calendar is generated for \getvariable{PocketDiary}{Place},
  \getvariable{PocketDiary}{Country}

```

```
\stopmode
\startmode[**de]
  Dieser Kalender ist berechnet für \getvariable{PocketDiary}{Place},
  \getvariable{PocketDiary}{Country}
\stopmode
% ...
\starttabulate[|l|r|]
  \NC Longitude
  \EQ \getvariable{PocketDiaryGeoPosition}{lon} \NC\NR
  \NC Latitude
  \EQ \getvariable{PocketDiaryGeoPosition}{lat} \NC\NR
\stoptabulate
\stopstandardmakeup

\getbuffer[Yearofmonthtables]

\stoptext
```

Und so sehen die Seiten aus, die dabei herauskommen:

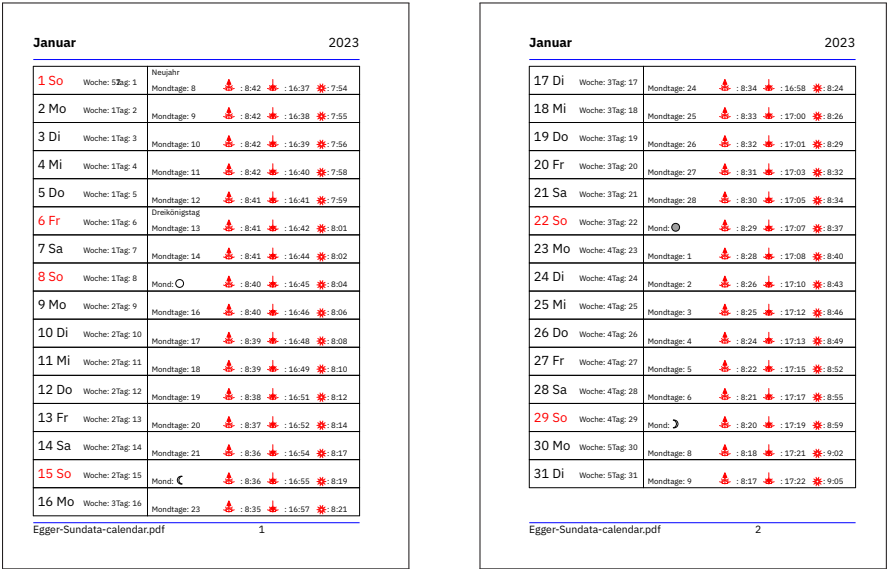


Abb. 5: Januar 2023

ConT_EXt kurz notiert

Henning Hraban Ramm

In der ConT_EXt-Welt ist vieles in Bewegung, das eine Erwähnung wert ist, aber keinen ganzen Artikel begründet. (Stand: 14. 4. 2023)

Matheschriften

Hans Hagen hätte die Unterstützung für alte (nicht-OpenType) Mathematikschriften gerne aus LMTX entfernt, damit wären aber die Mathe-Schriften von Antykwa Toruńska, Kurier und Iwona weggefallen, die nur als Type-1 vorliegen. Also hat er den Supportcode neu implementiert und verbessert, so dass diese exotischen Matheschriften jetzt ebenso gut unterstützt werden wie moderne OpenType-Fonts. Fehlende Symbole werden aus Latin Modern und den AMS-Schriften ergänzt.

Auch die von Daniel Flipo¹ gepflegten OpenType-Matheschriften wie Concrete, Erewhon und Euler werden jetzt durchgehend unterstützt.

Welche der Neuerungen aus dem Update vom 1. April² Wirklichkeit sind, ist schwer zu sagen: Darüber, die Unterstützung für Nicht-SI-Einheiten zu entfernen (»Niemand braucht Angaben in Zoll!«), witzeln wir schon lange. Dass ein paar Makros für Symbole wegfallen, die niemand verwendet, und dafür neue Einzug halten, die von Wissenschaftler*innen angefragt wurden, halte ich für realistisch. Dass ConT_EXt ein KI-System zur Interpretation natürlicher Sprache in der Mathe-Eingabe bekommt, eher weniger.

Gleitobjekte

Die Behandlung von Gleitobjekten (Floats) ist überarbeitet und erweitert worden, z. B. sorgt das Stichwort »always« bei bottom-Floats dafür, dass sie tatsächlich nur am unteren Rand des Satzspiegels platziert werden und nicht alternativ auf der nächsten Seite oben. Mehrstufige Ersatzplatzierungen (wenn a nicht geht, dann b, sonst c) werden möglich. Die Platzierung kann auch auf später verschoben und dann manuell ausgeführt werden, z. B. um verschieden große Bilder auf einer Seite anzuordnen. Die Dokumentation dazu steht im Handbuch »step by step«.

¹<https://ctan.org/author/flipo>

²<https://mailman.ntg.nl/pipermail/ntg-context/2023/108088.html>

Tabellenköpfe

Bei »natural tables« (HTML-Tabellen, `\bTABLE`) werden Tabellenköpfe (`\bTABLEhead`) bei Seitenumbrüchen schon immer wiederholt. Jetzt geht das zusätzlich auch mit Zwischenüberschriften (`\bTABLEsection`).

Dokumentübergreifende Referenzen

Noch in Arbeit, aber bei Erscheinen dieser Ausgabe wahrscheinlich verfügbar, ist die Erweiterung von dokumentübergreifenden Referenzen und Listen. Damit werden Inhaltsverzeichnisse, Register und Verweise in mehrbändigen Werken leicht realisierbar. ConT_EXt verarbeitet dabei die `.tuc`-Datei der anderen Dokumente und fasst die jeweiligen Tabellen zusammen.

Pandoc

Schon länger kann der Dokumentenkonverter auch ConT_EXt-Code ausgeben. Seit der Ende Januar freigegebenen Pandoc-Version 3³ haben die Entwickler (insbesondere wohl Albert Krewinkel) die Unterstützung für ConT_EXt stark verbessert:

- Syntax-Hervorhebung für Quellcode: wie bisher schon für HTML, L^AT_EX und andere Formate, mit Definitionen aus dem KDE-Projekt
- Inline-Code: konsistente Verwendung von `\type`
- Tagged PDF⁴ (z. B. PDF/A) mit dem Aufrufparameter `--to=context+tagging` in Kombination mit `-V pdfa=3a`
- Nicht nummerierte bzw. im Inhaltsverzeichnis ausgelassene Überschriften durch die Klassen `unnumbered` bzw. `unlisted`
- Abbildungen: bessere Unterstützung, auch für Unter-Abbildungen in Gleitobjekten

Als Eingabeformat für Pandoc kommt häufig Markdown zum Einsatz, aber auch Emacs Org-Mode hat engagierte Fürsprecher.

Chinesisch

Li Yanrui (李延瑞) hat sich um Korrekturen in der Unterstützung für Chinesisch bemüht und arbeitet an einer Anleitung, die im CTAN⁵ verfügbar ist.

³ <https://github.com/jgm/pandoc/releases/tag/3.0>

⁴ ConT_EXt-PDFs aus Pandoc sind immer »tagged«, die Parameter verbessern aber das Tagging.

⁵ <https://www.ctan.org/pkg/context-notes-zh-cn>

T_EXlive 2023

In der aktuellen T_EX-Live-Distribution ist die Standardversion von ConT_EXt zum ersten Mal LMTX (auf Basis von LuaMetaT_EX). MkIV ist nach wie vor enthalten, MkII ist weggefallen.

Von fremden Bühnen

Es gibt immer wieder Menschen und Projekte, die sich an »T_EX-Nachfolger« wagen; aufgefallen sind mir in letzter Zeit:

- Typst (Martin Haug und Laurenz Mädje),
<https://typst.app>;
»public beta«.
- Boxes and Glue (Patrick Gundlach),
<https://github.com/speedata/boxesandglue>;
Library in Entwicklung.
- TEI Publisher (e-editiones.org),
<https://tei-publisher.com> (für TEI, DocBook, JATS u. ä.);
im März wurde Version 8.0 veröffentlicht.

Die Spezifikation des PDF-2.0-Standards ist jetzt bei der PDF Association kostenlos verfügbar⁶. Die Dokumente müssen zwar für 0,00 € »gekauft« werden, es ist aber keine gültige Adresse nötig.

⁶ <https://www.pdfa.org/sponsored-standards/>

Von fremden Bühnen

Die Morgendämmerung der Strenge in der Kunst des Programmierens¹

Sudhir Rao, Indranath Sengupta

Donald Knuth ist emeritierter Professor für die Kunst der Computerprogrammierung an der *Stanford University*. Schon im Titel verbirgt sich der Schlüssel zu der Philosophie, die Knuths Herangehensweise an die Computerprogrammierung zugrunde liegt – dass es einer delikaten Mischung aus Kunstfertigkeit und analytischem Denken bedarf, um Werke zu schaffen, die den Test der Zeit bestehen. Ob es sich um seine Bestseller-Bände von *The Art of Computer Programming (TAOCP)* oder seine grundlegenden Beiträge zu »*The Analysis of Algorithms*« handelt, er hat häufig neue Wege beschritten; viele dieser Werke gelten heute als ideologisches Fundament, auf dem neuere Strukturen aufgebaut wurden. Dem umfassenden Gespräch mit Knuth, das von seinen Anfängen bis in die Gegenwart reicht, geht eine einleitende biografische Skizze unseres Protagonisten voraus.

Biografische Skizze

Donald Ervin Knuth bewundert eine große Sammlung von Skulpturen im Frogner-Park, Norwegens größter Touristenattraktion in der Hauptstadt Oslo und gerät dabei in eine sehr nachdenkliche Stimmung. Er befindet sich in einem Teil des Parks, der dem Leben und dem Werk des Künstlers Gustav Vigeland gewidmet ist. Alles ist geschmackvoll auf einer Fläche von 80 Hektar im Zentrum des Parks angeordnet. Wenn er über sein Leben und seine Kunst nachdenkt, weiß er instinktiv, wie tief befriedigend es für Vigeland gewesen sein muss, ein lebenslanges Streben zu einem erfüllenden Abschluss zu bringen. Ein Künstler kann sich nicht mehr wünschen, murmelt er vor sich hin.

¹ Zuerst erschienen in der Januarausgabe von Bhāvanā – The mathematical magazine (<https://bhavana.org.in>). Übersetzung mit Genehmigung von Bhāvanā durch deepL (<https://www.deepl.com/translator/>) und Herbert Voß.



Abb. 1: Vortrag an der Case Western Reserve University im Jahr 2010 (Dasha Slobozhanina)

Knuths Gedanken über Vigeland geben uns einen Einblick in einen kreativen Geist, der das Werk eines anderen sorgfältig versteht und schätzt. Auch Knuths eigene Beiträge zur Welt der Informatik wurden von Experten als bahnbrechend und wegweisend bezeichnet und haben den strengen Test der Zeit bestanden. Während die meisten anderen froh wären, wenn sie nur einen Bruchteil der Arbeit, die Knuth in den letzten sechs Jahrzehnten geleistet hat, für sich beanspruchen könnten, muss betont werden, dass Donald Knuth immer viele Ebenen über dem Gewöhnlichen gelegen hat.

Seiner Arbeit werden bahnbrechende Beiträge zugeschrieben, z. B. wie Software konzipiert, geschrieben, kompiliert und schließlich verarbeitet werden muss, und sie hat den gesamten Lebenszyklus der modernen Programmierung beeinflusst. Der Künstler in ihm hat sogar dafür gesorgt, dass das Endergebnis solcher Programme durch ästhetisch gestaltete typografische Schriftarten, die heute allgegenwärtig sind, ansprechend dargestellt wird. Indem er das, was er liebevoll die Kunst der Computerprogrammierung nennt, mit der Strenge der Mathematik verband, hat er ein blühendes Fach namens *Analyse von Algorithmen* geschaffen und damit den theoretischen Kern der Informatik beeinflusst. Letztere Errungenschaft ist für die theoretische Informatik nicht weniger bahnbrechend als die Erfindung des Mikroskops für die Biowissenschaften, des Teleskops für die Astronomie oder des Endoskops für die medizinischen Wissenschaften. Durch rigorose Analysen und durchdringende Einsichten hat Knuth tief und scharfsinnig in das vor ihm meist

undurchsichtige Innere eines Fachgebiets geblickt, das gerade erst aufkam; und damit hat er die Zwillinglandschaften des wissenschaftlichen Rechnens und der digitalen Technologie tiefgreifend beeinflusst. Sein Vortrag auf dem prestigeträchtigen *Internationalen Mathematiker-Kongress (ICM)* in Nizza, Frankreich, im Jahr 1970 trug den vorausschauenden Titel *The Analysis of Algorithms*. Damals war Knuth gerade einmal 32 Jahre alt, aber seine Arbeit fand bereits Anerkennung in den höchsten akademischen Foren, und die Einladung zu seinem Vortrag in Nizza war eine Bestätigung dafür, dass sein Arbeitsgebiet zwar das noch im Entstehen begriffene Fachgebiet der Informatik und ihrer Anwendungen war, seine Erkenntnisse aber bereits so streng und interessant waren, dass die anspruchsvolle Welt der Mathematik aufhorchte und Notiz nahm.

In den fünf Jahrzehnten seit Knuths ICM-Rede von 1970 hat sich die Unverzichtbarkeit der Strenge in der Informatik inzwischen gut etabliert. Ariele Iserles vom Fachbereich für Angewandte Mathematik und Theoretische Physik der *Universität Cambridge* sagte dies bei der Besprechung von Knuths *The Art Of Computer Programming, Vol 4* in der Juni-Ausgabe 2007 der *SIAM Review*:

Wir alle halten diese universelle Wirksamkeit der Mathematik für selbstverständlich. Dennoch ist es wichtig, hin und wieder innezuhalten und sich zu fragen, was genau die Mathematik zu einem solch universellen Werkzeug macht. Die Antwort liegt auf der Hand: Strenge. Es ist vielleicht nicht überraschend, dass Informatiker die größten Nutznießer mathematischer Strenge sind: Im Gegensatz zu proteinbasierten Lebensformen geben sich Computer nicht mit weniger zufrieden.

Knuth selbst hatte sich übrigens bereits drei Jahrzehnte zuvor, im Dezember 1974, in einem in der Zeitschrift *Communications of the ACM* veröffentlichten Artikel ähnlich geäußert:

Es ist sicherlich wünschenswert, die Computerprogrammierung zu einer Wissenschaft zu machen, und wir haben in den letzten fünfzehn Jahren tatsächlich einen weiten Weg zurückgelegt. Vor 15 Jahren verstand man so wenig von Computerprogrammierung, dass sich kaum jemand Gedanken darüber machte, wie man die Korrektheit von Programmen nachweisen konnte; man tüftelte einfach an einem Programm herum, bis man »wusste«, dass es funktionierte. Damals wussten wir noch nicht einmal, wie wir das Konzept, dass ein Programm korrekt war, in irgendeiner strengen Weise ausdrücken konnten. Erst in den letzten Jahren haben wir etwas über die Abstraktionsprozesse gelernt, mit denen Programme geschrieben und verstanden werden, und dieses neue Wissen über das Programmieren zahlt sich in der Praxis jetzt aus. Wenn wir heute Programme schreiben, wissen wir, dass wir im

Prinzip formale Beweise für ihre Korrektheit konstruieren können. Diese wissenschaftliche Grundlage führt zu Programmen, die wesentlich zuverlässiger sind als diejenigen, die wir früher geschrieben haben, als die Intuition die einzige Grundlage für die Korrektheit war.

Die Umstände, die Knuth aus dem kalten, idyllischen Wisconsin auf den sonnigen und pulsierenden Campus in Kalifornien brachten, wo die Computertechnik einige ihrer größten Sprünge machen sollte, sind eine faszinierende Geschichte für Neulinge und Kenner gleichermaßen. Donald Ervin Knuth wurde am 10. Januar 1938 in Milwaukee, Wisconsin, als Sohn eines lutherischen Ehepaars geboren, das in seiner Gemeinde, seinem Glauben und seiner Musik tief verwurzelt war. Als erstes von zwei Kindern hat Knuth alle drei elterlichen Eigenschaften geerbt und in sich aufgesogen – nämlich eine starke Gemeinschaft und Verbundenheit mit den Mitgliedern einer auserwählten Gemeinschaft, eine lebenslange Verwandtschaft mit der Kirche und eine anhaltende Liebe zur klassischen Instrumentalmusik: Dem Klavier und der Orgel, um genau zu sein. Als neugieriges Kind begann Knuth schon sehr früh zu lesen. Da er sich nicht besonders für Sport interessierte, boten ihm Bücher eine großartige Ablenkung. Er erinnert sich an einen Vorfall aus dem Jahr 1943, der sich in einer großen öffentlichen Bibliothek ereignete, die nicht weit von seinem Haus in Milwaukee entfernt war. An jenem Tag erreichte Knuth ohne Begleitung seiner Eltern und allein mit der Straßenbahn die Bibliothek und verlor sich völlig inmitten von Büchern, zusammengerollt mit einem Buch in einer gemütlichen Ecke. Versehentlich schloss der Bibliothekar die Bibliothek nach Feierabend ab, und der fünfjährige Knuth war immer noch drinnen, ohne dass beide Parteien etwas voneinander wussten. Eine verzweifelte Suche endete jedoch erfreulich: Knuths Eltern konnten ihn ausfindig machen, allerdings nicht ohne einen großen Schreck!

Neugierig und fleißig nahm er an Wettbewerben für Mittel- und Oberstufenschüler teil und gewann sogar einige davon. Diese Siege und die Anstrengungen, die er unternahm, um eine ganze Menge des erforderlichen außerschulischen Materials zu meistern, bereiteten ihn mental auf die bevorstehende Reise vor. Das *Case Institute of Technology*², wie es damals hieß, wurde ihm von seinen Familienmitgliedern wärmstens empfohlen, und Knuth begann dort sein Grundstudium zu absolvieren. Die Jahre, die er auf dem Case-Campus verbrachte, erwiesen sich als äußerst vorteilhaft, denn sie spornten ihn an, Mathematik und Informatik zu studieren, zwei Interessen, die in seinem Kopf heftig um die Vorherrschaft wetteiferten. Ernsthafte sportliche Aktivitäten waren auf dem Case-Campus ohnehin ausgeschlossen, da er generell kein Interesse an Sport hatte. Dennoch interessierte er sich für Sport auf eine rein akademische Weise. Als er als Manager der Basketballmannschaft von Case in der interuniversitären Liga arbeitete, stieß er auf ein interessantes

² Die heutige *Case Western Reserve University*.

Problem, bei dem es um die Leistungsprognose der Mannschaft und ihrer einzelnen Spieler ging. Knuth stürzte sich kopfüber in die Materie und stellte fest, dass man bis dahin nur die von den Spielern erzielten Punkte berücksichtigt hatte, nicht aber die anderen Dinge, die Basketballspieler tun. Ihm fiel auf, dass der Ballbesitz ein wichtiges Element war, das nie in die Statistik einging, ebenso wenig wie die Anzahl der *Fumbles*, *Steals*, *Rebounds* und Fehlwürfe. Damals notierte der Punktezüehler, wie viele Fouls ein Spieler begangen hatte, aber nur, weil niemand nach fünf Fouls weiterspielen konnte. Also beschloss Knuth, viel mehr Dinge zu zählen, als jemals zuvor gezählt worden waren. Nach jedem Spiel ging er zum Rechenzentrum und lochte Karten, um diese neumodischen Statistiken in den Computer auf dem Campus einzugeben. Der Computer setzte sie dann in eine Formel ein, die er entwickelt hatte, um den tatsächlichen Beitrag eines jeden Spielers zum Spiel zu schätzen. Trainer Phil Heim stimmte zu, dass diese neuen Zahlen »genau richtig« waren, und er forderte die Spieler auf, ihre berechneten Werte zu verbessern, anstatt einfach nur die meisten Bälle in den Korb zu werfen. Niemand hatte zuvor darauf hingewiesen, dass ein Spieler nicht wirklich zwei Punkte macht, wenn er punktet, weil er auch den Ballbesitz verliert. Das Ergebnis: Case wurde in jenem Jahr Meister der Liga.³ Diese Leistung erregte so viel Aufsehen, dass sie im nationalen Fernsehen und in der bekannten, international verbreiteten Zeitschrift *Newsweek* erwähnt wurde.⁴

Es gab noch einen weiteren Vorfall aus jener Zeit, bei dem es um die Lösung eines ziemlich schwierigen Problems ging, das von einem Professor gestellt wurde, dessen Kurse einen schlechten Ruf hatten. Louie Green stellte, wie es seine Gewohnheit war, Knuths Klasse ein charakteristisch schwieriges Problem und erklärte, dass jeder, der es lösen würde, für den Rest des Semesters nicht mehr zu seinen Vorlesungen erscheinen müsse. Außerdem erklärte Green, dass die richtige Lösung dem Studenten eine sehr begehrte Eins plus einbringen würde. Um es kurz zu machen: Knuth löste das Rätsel, erhielt eine Eins plus und schwänzte zum Glück auch den Unterricht. Übrigens hatte George Pólya selbst das gleiche Problem in einer damals aktuellen Ausgabe des *American Mathematical Monthly* gelöst. Das Schicksal wollte es, dass Knuth viel später sogar als Pólyas jüngerer Kollege in Stanford anfang und Pólyas Einfluss auf seine eigene Forschung und Lehre freudig anerkannte. Ein Großteil von Knuths intellektuellem Aufblühen auf dem Case-Campus wurde auch durch viele lange Stunden ermöglicht, die er an einem brandneuen IBM 650 verbrachte, der damals gerade am Case installiert worden war. Knuth war auch als Berater für eine Firma namens TRW tätig, um Compiler-Software zu schreiben, noch während er als Student eingeschrieben war.

³ Die folgende URL enthält einen Videolink mit Don Knuth, Phil Heim und ihrem Basketballteam: <https://www.youtube.com/watch?v=dhh8Ao4yweQ&t=4s>.

⁴ *Newsweek*: 53(1), (5. Januar 1959), Seite 63: »What's That About a Score Card? A Computer's the Thing«.

Ein Wendepunkt in seinem Studentenleben an der *Case University* trat ein, als er die Meinung des berühmten indischen Mathematikers Raj Chandra Bose⁵ einholte, der damals zu Besuch aus North Carolina kam. Auch Bose hatte Knuths Klasse in Case ein schwieriges Problem gestellt und mit Ausnahme von Knuth hatte der Rest der Klasse keine großen Fortschritte gemacht. Knuth hingegen löste es nicht nur, sondern schrieb auch ein Computerprogramm, um seine Lösung zu überprüfen, und ließ es sogar auf lokal verfügbaren Computern laufen. Ein schwer beeindruckter Bose, mit Knuth als Co-Autor, sorgte sogar dafür, dass die Lösung in einer Fachzeitschrift veröffentlicht wurde. Das Problem hatte mit der Idee eines so genannten lateinischen Quadrats der Größe 12×12 zu tun. Bose war, wie Kenner zweifelsohne wissen, einer der drei »Euler-Spoiler«, die alle eine langjährige Vermutung des legendären Leonhard Euler widerlegt hatten. Diese Leistung hatte das Trio R.C. Bose, S.S. Shrikhande und E.T. Parker aus der ruhigen Umgebung der akademischen Welt ins internationale Rampenlicht katapultiert, und der Durchbruch schaffte es sogar auf die Titelseite der *New York Times*. Bose verfügte zweifellos über Brillanz, Autorität und weltweites Prestige. Er erkannte auch instinktiv, dass auch Knuth eine große Bereicherung für die globale Kombinatorik-Gemeinschaft sein würde, und drängte ihn, ein Studium am Caltech aufzunehmen. Insbesondere riet er Knuth, eine Zusammenarbeit mit Marshall Hall in Erwägung zu ziehen, der zufällig auch der Doktorvater seines Mitarbeiters E.T. Parker war. Knuth befolgte den Rat von Bose, ging zum Caltech und erlangte Ende 1963 unter Halls Anleitung seinen Dokortitel.

Wenn er über diese entscheidenden Tage in seinem Leben nachdenkt und über die Faktoren, die ihn schließlich dazu brachten, die Mathematik zu verlassen, um sich der Informatik zu widmen, obwohl er einen Dokortitel in Mathematik am Caltech erworben hatte, bietet Knuth eine gut durchdachte Antwort. Laut Knuth lassen sich Menschen, die gut in Mathematik sind und diese betreiben, im Großen und Ganzen einem der vier Typen mathematischer Begabung zuordnen:

- (i) algebraisch,
- (ii) geometrisch,
- (iii) logisch-kombinatorisch und
- (iv) allgemein.

Knuth fühlte sich immer am wohlsten bei der Verfolgung der logisch-kombinatorischen Aspekte. Dies, sagt er, lässt sich leicht mit den diskreten Problemen verbinden, die man normalerweise in der Informatik sieht, und diese innere Berufung beeinflusste schließlich seine Entscheidung, die mathematischen Aspekte der Informatik zu studieren. Es gab auch noch andere Beispiele, erinnert er sich. Einer davon eignete sich auf einer Seereise während seiner Flitterwochen. Während seine Frau

⁵ »Raj Chandra Bose: Universal Mathematician«, von H. Howard Frisinger und Indranath Sengupta. *Bhāvanā*, Vol 6(2) April 2022. <https://bhavana.org.in/raj-chandra-bose-universal-mathematician/>

vorübergehend unpässlich war, beschäftigte sich Knuth mit Noam Chomskys Buch *Syntactic Structures*. Als er das Buch zu Ende gelesen hatte, war er völlig davon überzeugt, dass Mathematik und Computertechnik eng miteinander verbunden sind. Andererseits war der Begriff Informatik noch nicht erfunden worden und die Forschung in diesem Bereich war nur ein Randthema, das weitgehend unerforscht war.

Ein kurzer Überblick über das noch junge Fach Informatik Mitte bis Ende der 60er Jahre zeigt uns, dass Computerhardware bei weitem nicht so allgegenwärtig war wie heute, und dass es nur eine Handvoll großer proprietärer Maschinen gab, die kommerziell erhältlich waren. Nicht nur die Hardware steckte noch in den Kinderschuhen. Die Software war noch weniger entwickelt und die entsprechende Situation in der Wissenschaft war nicht sehr ermutigend. Abgesehen von einigen wenigen fortschrittlichen Universitäten in den USA konnten die meisten Universitäten nicht einmal einzelne Kurse anzubieten, geschweige denn ausgebildete Dozenten mit vollwertigen Abteilungen unterhalten. Nur Stanford und das MIT ragten wie Leuchttürme aus der gesamten Universitätslandschaft heraus; Stanford konnte sich 1965 mit der ersten Abteilung in den gesamten Vereinigten Staaten rühmen, die sich ausschließlich mit allen Aspekten der Informatik befasste.

Knuth begann seine akademische Laufbahn als Assistenzprofessor für Mathematik am Caltech, seiner Alma Mater, an der er promovierte. Formal war er ein ausgebildeter Mathematiker mit einer Spezialisierung auf kombinatorische Analyse. Es machte ihm Spaß, traditionelle Mathematikurse zu unterrichten, vor allem über diskrete Themen, aber seine Aufmerksamkeit richtete sich zunehmend auf Themen, die sowohl vom Geist als auch vom Wesen her eindeutig rechnergestützt waren. Er verbrachte viel Zeit damit, die *Burroughs Corporation* bei der Entwicklung von Software und Hardware zu beraten. Er wurde auch Mitherausgeber mehrerer Computerzeitschriften. Er führte Kurse über Computer in den Lehrplan von Caltech ein. Schließlich erkannte er, dass sein kreativer Schwerpunkt darin bestand, die quantitativen Aspekte der Algorithmenleistung zu untersuchen.

Im Hinblick auf die Neuausrichtung seiner philosophischen Einstellung zur Forschung, die seinem späteren Berufswechsel vorausging, stellt Knuth fest, dass es praktische Vorteile hat, sich jedem wissenschaftlichen Thema zu nähern, indem man seine rechnerischen und algorithmischen Aspekte genau betrachtet. Dieser Ansatz hat seiner Meinung nach eine tiefere Bedeutung, denn er glaubt, dass ein Mensch etwas erst dann wirklich versteht, wenn er es einem Computer beibringen kann, und unterstreicht damit die Bedeutung eines algorithmischen Ansatzes für die Beherrschung eines unbekannten Themas. Ein Beispiel, das er gerne anführt, stammt aus der Linguistik, wo er sagt

Linguisten dachten, sie würden Sprachen verstehen, bis sie versuchten, sie Computern zu erklären; sie lernten bald, wie viel mehr noch zu lernen ist.

Ein weiteres überzeugendes Beispiel stammt aus seiner Zeit als Mathematiker am Caltech:

Drei Jahre lang unterrichtete ich im zweiten Studienjahr einen Kurs in abstrakter Algebra für Mathematikstudenten am Caltech und das schwierigste Thema war immer das Studium der kanonischen Jordan-Form für Matrizen. Im dritten Jahr versuchte ich einen neuen Ansatz, indem ich das Thema algorithmisch betrachtete, und plötzlich wurde es ganz klar. Das Gleiche geschah mit der Diskussion endlicher Gruppen, die durch Generatoren und Relationen definiert sind, und in einem anderen Kurs mit der Reduktionstheorie binärer quadratischer Formen. Indem man das Thema in Form von Algorithmen darstellte, wurden der Zweck und die Bedeutung der mathematischen Theoreme transparent.

So gesehen ist die Entwicklung von Knuths Schwerpunkt in den 60er Jahren typisch für jeden Pionier, der sich zwar auf ein vielversprechendes neues Terrain wagt, sich aber auch der bevorstehenden Herausforderungen bewusst ist. Zwei Ereignisse aus dieser Zeit erweisen sich rückblickend als bedeutsam. Erstens: 1969 unternahm Knuth einen kühnen Vertrauensvorschuss und wechselte an das Department of Computer Science in Stanford, womit er unmissverständlich zu erkennen gab, wo seine akademischen Interessen lagen. Zweitens: Auf der ICM, die 1970 in Nizza, Frankreich, stattfand, präsentierte er einen Vortrag mit dem Titel »*The Analysis of Algorithms*«, der ihn auf den Weg brachte, mathematische Strenge und Präzision in die Formulierung, den Entwurf und die Analyse von Algorithmen einzubringen. Sein Kollege Robert W. Floyd hatte ihm Anfang der 60er Jahre beigebracht, dass die Informatik unbedingt eine solide mathematische Grundlage braucht.

Heute ist Knuths Name ein Synonym für mehrere Bände seines berühmtesten Werks: *The Art of Computer Programming (TAOCP)*. Die Ereignisse, die zum Verfassen des Buches führten, geben einen Einblick in den hervorragenden Ruf, den Knuth in akademischen und industriellen Kreisen genießt, und den er schon genoss, als er noch nicht einmal ein Doktorand war. Knuth erhielt vom Verlag Addison-Wesley ein Angebot, ein Buch über Computerprogrammierung zu schreiben. Das Angebot kam von Richard Varga, einem Mathematiker an der *Case University*, der auch im Beirat von Addison-Wesley saß. Er war damals 24 Jahre alt und als Doktorand am Caltech eingeschrieben. Varga hatte einen guten Eindruck von Knuths Fähigkeiten beim Schreiben von Compiler-Software. Knuth fühlte sich geehrt und war begeistert, denn viele seiner Lieblingslehrbücher für das College waren bei Addison-Wesley erschienen, und er hatte schon immer gerne geschrieben. In den frühen 60er Jahren

arbeitete er also tagsüber als Doktorand in Mathematik am Caltech und abends und nachts als Autor an einem Lehrbuch über Software. Außerdem war er als Berater für die Industrie tätig, wann immer er einen Moment Zeit hatte. Das bedeutete auch, dass er zwei prestigeträchtige Graduiertenstipendien ablehnen musste, eines von der *National Science Foundation* und ein weiteres von der *Woodrow Wilson Foundation*, weil er bei Annahme eines dieser Stipendien neben seinem Studium nichts mehr hätte tun können.

1962 war das *TAOCP* zunächst nur als ein einziges Buch mit genau zwölf Kapiteln geplant. Doch als er mit dem Schreiben begann, wurde ihm klar, dass die Themen, die er behandeln wollte, viel mehr Aufmerksamkeit und Details erforderten. Heute ist *TAOCP* kein einzelnes Buch mehr, sondern ein aufwendiger und üppiger Strauß der erlesensten Angebote der Informatik, verteilt auf fünf enzyklopädische Bände. Der jüngste, Band 4B, wurde erst im Oktober 2022 veröffentlicht. Er hofft, Band 4C im Jahr 2025 fertigstellen zu können, wenn er dann rüstige 87 Jahre alt sein wird, entschlossen und unermüdlich.

Während des Schreibens des ersten Bandes von *TAOCP* lernte Knuth den gesamten Lebenszyklus der Produktion eines Buches kennen. Knuth gibt stolz zu, »Tinte« im Blut zu haben, eine Anspielung auf Druckertinte. Sein Vater, der mehrere Jobs hatte, um die Familie in Milwaukee zu ernähren, besaß einen Vervielfältigungsapparat, und er vermutet, dass dies vielleicht unbewusst seine späteren Gedanken über digitale Typografie, Druckästhetik und Schriftgestaltung für die Verwendung in Computern beeinflusst haben könnte. Seine eigene typografische Reise begann völlig ungeplant mit einem starken Gefühl der Unzufriedenheit darüber, wie digitale Zeichen auf die Seiten eines Buches gedruckt wurden. Pat Winston, ein Computerwissenschaftler am MIT, hatte ein Buch über künstliche Intelligenz geschrieben, das als erstes mit einem experimentellen Digitaldruckgerät gesetzt wurde, und Knuth sah zufällig die Korrekturfahnen. Die Schriftqualität von Winstons Probedrucken war, obwohl sie mit rein digitalen Methoden hergestellt wurden, genauso gut wie die, die Knuth jemals zuvor mit konventioneller metallurgischer oder optischer Technologie gesehen hatte. Im Vergleich dazu war die Qualität des Fotosatzes in den aktuellen Fahnenabzügen von Knuths eigenem *TAOCP* miserabel; er wollte einfach nicht der Autor eines Buches sein, das so schrecklich aussah.

Heute ist die Computerwelt für diese innere Unruhe sehr dankbar. Wenn sich heutige Computeranwender und Industriedesigner an einem Übermaß an Schriftarten, Stilen und einer Vielzahl anderer typografischer Angebote erfreuen, die dazu beigetragen haben, ästhetisch gedrucktes digitales Material zu produzieren, so lässt sich vieles davon auf die lange Buße zurückführen, die Knuth auf sich nahm und damit die Grundlagen der digitalen Typografie beeinflusste. Ein Bericht über die Anfänge der gesamten Typografie-Episode in seinen eigenen Worten zeigt den

missionarischen Eifer, mit dem er sich im Allgemeinen bedeutsamen Herausforderungen stellte, immer bewaffnet mit der stillen Zuversicht, dass sich seine Beiträge schließlich als bedeutend erweisen würden. Als er die Seiten von Winstons Druckfahnen durchblätterte, sagte Knuth mit einer für ihn typischen Kombination aus Ungläubigkeit und Selbstvertrauen:

Ich dachte, Pixel könnten es einfach nicht schaffen. Aber hier, vor meinen Augen, war ein Beispiel von höchster Qualität und ich wusste, dass alles, was ich auf dieser Seite sah, von 0en und 1en erzeugt wurde. Es wurde nicht durch einen geheimnisvollen Metallprozess, einen mysteriösen fotografischen Prozess oder irgendetwas, das mir verborgen oder unheimlich war, erzeugt. Es war digital und doch wunderschön. Am nächsten Morgen wachte ich auf und wusste, dass sich mein Leben verändern würde. Wow! Ich stand vor dem großen Problem, wie ich meine Bücher richtig aussehen lassen konnte, aber jetzt hatte ich den handfesten Beweis, dass die Lösung dieses Problems in meiner Macht lag. Alles, was ich tun musste, war, einen Weg zu finden, 0en und 1en auf ein Raster zu setzen, wobei die 1en für Tinte standen - und wissen Sie, ich bin ein ziemlich eingebildeter Kerl, und ich glaube, ich kann 0en und 1en so gut wie jeder andere auf der Welt. Und nicht nur das. Weil ich gut in 0 und 1 war und weil 0 und 1 die Zukunft des Drucks sein sollten, war es meine Aufgabe, darüber nachzudenken, wie man diese 0 und 1 dorthin bringen konnte. Auch andere Leute mussten die digitale Typografie nutzen. Als Informatiker musste ich an der digitalen Typografie arbeiten, um dieses Problem zu lösen, denn es behinderte nicht nur mich, sondern auch viele andere.

Knuths Beschäftigung mit seiner Arbeit und seinen Büchern ist ein perfektes Beispiel für die unaufhörliche Besessenheit eines unnachgiebigen Meisters von seinem Handwerk und seinen eigenen esoterischen Werkzeugen. Wenn er neue Bände für *TAOCP* schreibt, veröffentlicht er das Material zunächst in kleineren Taschenbuch-Faszikeln, die einem bestimmten fortgeschrittenen Thema gewidmet sind. Und während er diese Faszikel vorbereitet, stellt er Vorab-Faszikel ins Internet, damit Experten auf dem Gebiet ihm helfen können, die Geschichte richtig zu schreiben. Zu seiner ständig wachsenden Schar von Bewunderern und Anhängern gehören Hunderte von Freiwilligen, die ihm helfen, Fehler zu korrigieren, bevor die eigentlichen Faszikel und Bände veröffentlicht werden. Neben den Büchern, die er verfasst hat, hat er auch die weit verbreitete Computersprache $\text{\TeX}$ entwickelt – das »Brot und die Butter« für Menschen in der akademischen Welt und in der Industrie. Für eine dankbare weltweite Computergemeinde sind seine Werke wie kostbare Edelsteine, die von einem seltenen Kunsthandwerker, der unermüdlich an seinen

Konstruktionen meißelt, sorgfältig an Ort und Stelle verziert wurden, während sie sich entwickeln und Gestalt annehmen.

Es versteht sich von selbst, dass er einen straffen Zeitplan hat und an vielen Projekten gleichzeitig arbeitet. Da er fast vollständig in seine Arbeit vertieft ist, ist die Musik eine wichtige Nebenbeschäftigung. In seinem zum Büro umfunktionierten Haus stehen eine Pfeifenorgel und ein großes Klavier, die ihm sowohl Erholung als auch Inspiration bieten. Obwohl er sich offiziell aus Stanford zurückgezogen hat und nicht mehr verpflichtet ist, Studenten zu unterrichten, hält er gelegentlich noch Vorträge und Vorlesungen in der akademischen Gemeinschaft. Eine weitere Eigenart Knuths, an die sich die Computerwelt nur widerwillig gewöhnt hat, ist seine selbst auferlegte vollständige Isolierung von E-Mails. Er kann nicht direkt per E-Mail erreicht werden und hat ein ziemlich ausgeklügeltes Protokoll für Leute eingerichtet, die mit ihm kommunizieren wollen. Aber das hält den unbezähmbaren Knuth nicht davon ab, E-Mails an andere zu verschicken, wobei den Empfängern dieser seltenen und unerwarteten Kommunikation von Anfang an klar ist, dass der Datenverkehr nur in eine Richtung geht und ausschließlich von Knuth allein kontrolliert wird! Die restliche Korrespondenz, die ihn schließlich erreicht, wird auf dem Postweg verschickt, wobei seine Sekretärin in Stanford dafür verantwortlich ist, das, was er zu lesen bekommt, zu sichten und in die engere Auswahl zu nehmen. Diese Ausgabe von Bhāvanā enthält ein Interview mit Knuth und ist ihm zu seinem 85. Geburtstag gewidmet.

Ursprünglich für eine Ausgabe Ende 2021 geplant, freuen wir uns, dass wir nun endlich das dazugehörige Interview ans Tageslicht bringen können. Der stets lebenswürdige und hilfsbereite Gentleman Knuth hat sich unsere Fragen durchgelesen und mit charakteristischem Elan eine herzliche Antwort gegeben. Alle seine Antworten auf alle Fragen, die wir ihm gestellt haben, sind genau 280 Zeichen lang und zeigen einmal mehr, warum die Welt sein einzigartiges geistreiches Genie feiert. Der sympathische Professor entschuldigte sich in einer anerkennenden Notiz auch für die einjährige Verspätung und machte sie für unsere Leser mit seinem Intellekt und seiner Begeisterung mehr als wett.

Es gibt kaum einen Preis in der Computerwissenschaft, den Knuth nicht gewonnen hat. Er wurde 1971 mit dem *Grace Murray Hopper Award*, 1974 mit dem *Turing Award*, 1979 mit der *National Medal of Science*, 1995 mit der *John von Neumann Medal* und 2011 mit der *Faraday Medal* ausgezeichnet, außerdem erhielt er Mitgliedschaften und Stipendien vieler gelehrter Gesellschaften der Welt. Im Jahr 1996 wurde ihm der *Kyoto-Preis* verliehen, der mit einem Preisgeld von 450.000\$ versehen war, inflationsbereinigt fast 800.000\$ im Jahr 2023. Der gesamte Betrag wurde an vier Wohltätigkeitsorganisationen gespendet, die dem Ehepaar Knuth sehr am Herzen lagen, was zeigt, wie hoch sein Ansatz immer war.

Bis zum heutigen Tag widmet sich Knuth mit großer Leidenschaft der historischen Erforschung aller Aspekte der Computertechnik. Datenstrukturen, die als Bäume bekannt sind, drücken Beziehungen zwischen kaskadierenden Knoten aus und erfassen effektiv Genealogien, Einflüsse, Abhängigkeiten sowie historische Hinterlassenschaften. Übrigens wurden Bäume von Knuth ausgiebig untersucht. Ariele Iserles wiederum hat in seinem bereits zitierten Artikel in der *SIAM Review* Folgendes über Knuths Behandlung einiger historischer Aspekte der Informatik gesagt:

Wenn wir Bäume nicht nur als abstrakte kombinatorische Objekte betrachten, sondern als ein Mittel, um kombinatorische Beziehungen auszudrücken, berühren sie Themen, die Gelehrte seit Jahrtausenden beschäftigen: von chinesischen Fliesenarrangements über indische und griechische poetische Metren bis hin zur mittelalterlichen Kabbala und mittelalterlichen Musik und darüber hinaus. Das Streben nach Struktur, Muster und Symmetrie liegt der menschlichen Kultur und Wissenschaft zugrunde, auch wenn es sich manchmal in den Versen der Dichter und nicht in mathematischem Formalismus ausdrückt.

Interessanterweise hatte Knuth vor einem halben Jahrhundert, im Jahr 1972, babylonische Tafeln aus der Zeit zwischen 1800 und 1600 v. Chr. untersucht, um zu verstehen, wie einige ihrer antiken Inhalte verstanden werden könnten, wenn man sie durch die Linse der modernen Computertechnik betrachtet. Wenn man sich einige seiner Ausführungen genau ansieht, wird deutlich, dass Knuth eigentlich für das Gegenteil der obigen Aussage plädiert:

Eine der Möglichkeiten, der Informatik zu Ansehen zu verhelfen, besteht darin zu zeigen, dass sie tief in der Geschichte verwurzelt ist und nicht nur ein kurzlebiges Phänomen darstellt. Daher ist es nur natürlich, sich den frühesten überlieferten Dokumenten zuzuwenden, die sich mit dem Rechnen beschäftigen, und zu untersuchen, wie die Menschen vor fast 4000 Jahren an das Thema herangegangen sind.

Diese Bemerkung aus dem Jahr 1972 ist historisch von unschätzbarem Wert, denn sie zeigt, wie Pioniere wie Knuth noch vor fünfzig Jahren unermüdlich darum bemüht waren, einer damals noch jungen Disziplin, die heute allgegenwärtig und zu einem unverzichtbaren Teil unseres gesellschaftlichen Gefüges geworden ist, Seriosität und Glaubwürdigkeit zu verleihen.

Um Knuths Philosophie der Kreativität zu verstehen, ist ein von ihm bewundertes Zitat aus dem Buch *The Science of Art* von Robert E. Mueller besonders relevant:

Früher dachte man, dass die phantasievolle Perspektive des Künstlers der Tod für den Wissenschaftler sei. Und die Logik der Wissenschaft

schien allen möglichen künstlerischen Fantasieflügen zum Verhängnis zu werden.

Knuth fügt hinzu, dass der wissenschaftliche Ansatz im Allgemeinen durch die Worte logisch, systematisch, unpersönlich, ruhig und rational charakterisiert wird, während der künstlerische Ansatz durch die Worte ästhetisch, kreativ, humanitär, ängstlich und irrational gekennzeichnet ist. Für ihn haben diese beiden scheinbar widersprüchlichen Ansätze einen großen Wert in Bezug auf die Computerprogrammierung. In Knuths Rechenuniversum können Computerprogramme, die aus der Synthese der beiden Ansätze entstehen, zwar elegant, exquisit und vielleicht manchmal sogar funkelnd sein, aber man muss sich ständig bemühen, Programme zu schreiben, die großartig, edel und wahrhaft prächtig sind!

Und schließlich ist es ein Zeichen für die Größe eines Menschen, unabhängig vom Bereich, in dem er tätig ist, wenn immer wieder Anspielungen auf einen Platz in den Annalen der Geschichte gemacht werden, selbst wenn der Protagonist nachdenklich und mit Nonchalance an seiner Werkbank hämmert. Das unablässige Streben des Geistes, eine höhere Ebene zu erreichen, ist unser wahres kollektives Erbe des *Don of the Art of Computer Programming*!

Interview

Im Namen des Teams Bhāvanā und des Projekts *History of Mathematics in India (HoMI)* des IIT Gandhinagar ist es uns eine Freude und ein Privileg, Sie einzuladen, einen persönlichen Bericht über Ihr ungemein produktives Leben und Ihre bahnbrechenden Arbeiten auf dem Gebiet der Informatik mit uns zu teilen. Wir möchten diesen Artikel, der in dieser Ausgabe erscheint, als unseren bescheidenen Tribut und als Geschenk zu Ihrem 85. Geburtstag widmen.

DK: Ich fühle mich zutiefst geehrt, dass Sie mich interviewen möchten, denn ich liebe den Geist Ihrer Zeitschrift. Ich wünschte, es hätte ein ähnliches Magazin gegeben, das solche Interviews mit (sagen wir) Euler, Gauß und Ramanujan geführt hätte, als sie noch am Leben waren! Ich komme sicher nicht annähernd in die Fußstapfen eines dieser Giganten, aber ich glaube fest daran, dass jeder durch das Studium der Erkenntnisse kreativer Menschen lernen kann.



Abb. 2: Der 2-jährige Don am Klavier, auf dem sein Vater seine Schüler zu unterrichten pflegte (1940 Ervin Knuth).

Ich sehe, dass Sie mir Fragen stellen werden, die auf Dingen basieren, die Sie bereits über mein Leben erfahren haben. Darüber ist offensichtlich schon viel geschrieben worden; daher halte ich es für das Beste, wenn ich mich an die weise Politik von Twitter halte, die Länge meiner Antworten zu begrenzen. Daher werde ich mein Bestes tun, um jede Ihrer Fragen mit einem »Tweet« von genau 280 Zeichen zu beantworten.

Wir haben erfahren, dass Sie deutsche Vorfahren haben und dass Sie ein ziemlich frühreifes Kind waren. Erzählen Sie uns etwas über Ihre Kindheit und Ihre Familie, und auch darüber, wie Ihr Name ausgesprochen wird.

DK: Mein Vater Ervin Knuth (NOOTH) war Lehrer an einer evangelischen Schule. Meine Mutter Louise verwaltete Gebäude in der Innenstadt von Milwaukee. Ich mochte Musik und Grammatik, schrieb aber auch abgedroschene Witze. In Sport und Kunst war ich schlecht. Habe viele Stunden damit verbracht, Diagramme zu zeichnen. Ich lernte, viele Bäume an ihren Blättern zu erkennen.



Abb. 3: An der Schreibmaschine seiner Eltern im Alter von 3 Jahren; hier zeigt sich eine frühe Faszination für Tastaturen (1941 Ervin Knuth).

Als Sie in den Boomjahren nach dem Zweiten Weltkrieg aufwuchsen, hatten Sie Zugang zu hervorragenden öffentlichen Schulen und Gemeindebibliotheken. Haben Sie sich damals in Büchern verloren, und vor allem, gab es in diesen Jahren positive Einflüsse und Ereignisse, sowohl zu Hause als auch außerhalb, die Ihre Jugend und Ihr junges Erwachsenenalter geprägt haben?

DK: Mit 4 Jahren war ich der jüngste »Bücherwurm« in der Bibliothek von Milwaukee. Einmal habe ich mich in der Biblio-

thek verirrt, weil ich über die Schließzeit hinaus in den Regalen blieb. Ich besuchte eine kirchliche Schule, wo meine Lehrer uns lehrten, andere zu lieben. Viele Gelegenheiten zu singen und zu musizieren, die Natur zu genießen. Ich hatte keine Ahnung von Mathe.

Eine Fernsehsendung, die ausgestrahlt wurde, als Sie noch zur Schule gingen, schlug eine wichtige Brücke, die die Satzstruktur der Sprache mit Diagrammen verband und Ihr lebenslanges Verständnis für Sprachen und ihre innere Struktur tief beeinflusste. Bitte erläutern Sie dieses einschneidende Ereignis.

DK: Meine Lehrerin in der 7. Klasse führte uns in das Diagrammieren von Sätzen ein. Meine Freunde und ich versuchten, sie auf Beispiele anzuwenden, die nicht aus dem Lehrbuch stammten, mit begrenztem Erfolg, aber wir lernten dabei eine Menge. Die Fernsehsendung war anders: Ich gewann einen Wettbewerb, bei dem es darum ging, so viele Wörter wie möglich aus vorgegebenen Buchstaben zu finden.

Nach dem Zweiten Weltkrieg machten die Amerikaner nicht nur enorme Fortschritte in der Wissenschaft, sondern bauten auch Computermaschinen, die von der Stange erhältlich waren. Wann haben Sie zum ersten Mal von einem Digitalcomputer gehört, ihn in die Hand genommen und schließlich selbst programmiert? War Ihnen damals schon klar, dass dies eine so intensive und leidenschaftliche Reise werden würde?

DK: Ich habe wahrscheinlich in der Wahlnacht 1952 zum ersten Mal vom UNIVAC gehört, als ich 14 Jahre alt war. Einen echten Computer – den IBM 650 – sah ich zum ersten Mal 1956, als Studienanfänger am *Case College*. Wir durften diese Maschine anfassen, an der Konsole sitzen und sie mit Karten füttern. Ich war für immer süchtig danach.

Lassen Sie uns ein wenig mehr über Ihr Leben am ehemaligen *Case Institute of Technology*, der heutigen *Case Western Reserve University*, erfahren. Sie kamen mit 17 Jahren dorthin, und mit 22 hatten Sie Ihren Bachelor und Ihren Master gemacht. Welche Fächer haben Sie während dieser aufregenden und arbeitsreichen Zeit am *Case Institute* studiert?

DK: Erstsemester (Physik, Chemie, Kalkül, Staatsbürgerkunde, Schreiben); Zweitsemester (Astronomie, Grundlagen der Mathematik, Geometrie, Physik2, Geschichte, Sprechen); Nebensemester (Algebra, Topologie, Elektrotechnik, Literatur, numerische Analyse); Hauptsemester (Automaten, Kombinatorik, Logik, komplexe Variablen).

Neben Ihren herausragenden Studienleistungen haben Sie eine Studentenzeitschrift herausgegeben, probabilistisch begründete Algorithmen zur Bewertung von Basketballspielern anhand von Gewinnwahrscheinlichkeiten entwickelt und sogar eine IBM-Maschine auf dem Campus programmiert. Führen Sie uns bitte durch diese vier intensiven Jahre an der *Case University*.

DK: Ich marschierte in der Band, redigierte die Zeitung, war Vizepräsident einer Studentenverbindung, verliebte mich, leitete Sportteams, schrieb Compiler und Assembler, nahm an Mathematikwettbewerben teil, redigierte die Zeitschrift und das Studentenhandbuch, schaute bei Theaterstücken und Orchesterproben zu und schrieb eine kurze Musikkomödie.

Das *California Institute of Technology (Caltech)* hatte in den 60er Jahren den brillanten Richard Feynman und Murray Gell-Mann in der Physik. In der Mathematik gab

es das Ehepaar Jack Todd und Olga Taussky-Todd, die beide ein starkes Interesse an Aspekten der Informatik hatten. Als es jedoch an der Zeit war, über einen Dokortitel nachzudenken, entschieden Sie sich für Mathematik am Caltech, und zwar bei Marshall Hall Jr. Wie kam es zu dieser Wahl der Universität und des Betreuers, und worum ging es in Ihrer Doktorarbeit?

DK: R. C. Bose lehrte mich Kombinatorik und inspirierte mich, mit Hall zu arbeiten. Ich plante zunächst, Entwürfe mit $\lambda = 2$. Aber eines Tages konstruierte ich zufällig neue Arten von nicht-Desargues'schen projektiven Ebenen und löste damit eine Vermutung, und Marshall sagte mir, dass dies meine Diplomarbeit sein sollte.

Ihre Zeit als Doktorand war offenbar schon durch Ihre beiden konkurrierenden Interessen, Informatik und Mathematik, geteilt. Wir haben gehört, dass Sie in jenen Tagen auch als Berater für die Industrie tätig waren und in den Sommerferien durch Ihre Beratungstätigkeit sogar deutlich mehr verdienten als die Jahresgehälter der frischgebackenen Assistenzprofessoren! Hatte Hall Verständnis für diese Doppelfunktion, die Sie damals ausübten? Und warum haben Sie später beschlossen, die Beratung ganz aufzugeben?

DK: Ja, Hall glaubte schon früh an die Macht des Computers, um die kombinatorische Theorie zu entwickeln. Die Beratung war meine Verbindung zu dem neomodischen Gebiet der Informatik. Ich habe am Caltech Informatik gelehrt, sogar als Mathematikprofessor. Aber ich hörte mit der Beratung auf, als ich in Stanford CS-Professor wurde.

Etwa zur gleichen Zeit, als Sie einen Vertrag als Autor eines Buches über Computerprogrammierung erhielten, kam es zu einer Neuausrichtung Ihrer akademischen Interessen, die Ihnen viel später einen schärferen Fokus und eine neue Identität in der akademischen Welt verleihen sollte. Bitte führen Sie uns durch diese Phase Ihrer Karriere und die wichtigsten persönlichen Meilensteine aus dieser Zeit.

DK: Ein Vertreter von Addison-Wesley, dem Verlag meiner Lieblingslehrbücher, traf sich im Januar 1962 mit mir und lud mich ein, ein neues Buch über das Schreiben



Abb. 4: Eltern Ervin und Louise, mit Don und Schwester Paula (Mai 1942 Dorothea Eckart).

von Compilern zu schreiben. Das war aufregend, denn die vorhandenen Veröffentlichungen waren schlecht, einseitig und oft widersprüchlich. Ich hatte schon immer gerne geschrieben.



Abb. 5: Vorbereiten von Lochkarten für die IBM-Maschine in Case (1958 Jill Carter).

Nach Ihrer Promotion traten Sie als Assistenzprofessor in die Abteilung für Mathematik am Caltech ein. Aber tief in Ihrem Inneren waren Sie unruhig und zerbrachen sich ständig den Kopf darüber, dass Sie nicht in der Lage waren, die verschiedenen Wege zu beschreiten, die sich in Ihrem Kopf in Bezug auf die Informatik aufgetan hatten und die inzwischen auch zu einer intellektuellen Erregung geworden waren, die man nicht ignorieren konnte. Warum, wie und seit

wann hat die Informatik begonnen, fast alle Ihre ernsthaften akademischen Interessen in Anspruch zu nehmen?



Abb. 6: An der Konsole des IBM 650 in Case, mit dem Basketballtrainer Heim. (1960 Case News Service).

DK: Im Sommer 1962 schrieben mein Freund Bill Lynch und ich einen FORTRAN-Compiler für UNIVAC SS80-Computer. Eines Tages beschloss ich zu erforschen, warum sein »Hash-Algorithmus« gut funktionierte, und hatte Glück: Ich sah, wie dieses Problem zu lösen war, und erkannte, dass es viele ähnliche, bisher ungelöste Probleme gab.

Caltechs eigene, überraschend primitive Computerinfrastruktur in den 60er Jahren stand im Gegensatz zu den hervorragenden Computereinrichtungen in den *Jet Propulsion Laboratories (JPL)* nebenan. War dies im Großen und Ganzen auch symptomatisch dafür, dass Caltech damals etwas hinterherhinkte, weil es sich auf die Fortschritte im Bereich der Informatik konzentrierte, die anderswo stattfanden? War dies einer der Auslöser für Ihren späteren Weggang von dort?

DK: Als Berater hatte ich leichten Zugang zu hervorragenden Burroughs-Maschinen in der Nähe meines Wohnorts. Fast jede Universität hatte damals primitive Re-

chenanlagen. Stanford war eine Ausnahme: George Forsythe verstand, dass die Informatik ein neues Gebiet mit großen intellektuellen Herausforderungen war.

Volker Strassen, Stephen Cook und Sie waren drei junge Mathematiker, die zwischen Ende der 60er und Mitte der 70er Jahre wesentlich zu unserem heutigen Verständnis der Komplexität und Analyse von Algorithmen beigetragen haben. Während sich Strassen auf die Komplexität der linearen Algebra und der Matrixoperationen konzentrierte, war Cooks Arbeit eher auf die logischen und theoretischen Aspekte ausgerichtet. Wie sind Sie dazu gekommen, Ihren eigenen Weg einzuschlagen, der sich so sehr von dem Ihrer Zeitgenossen wie Victor Pan, James Wilkinson, Shmuel Winograd und anderer unterscheidet, die sich mit Problemen der numerischen Analyse und des wissenschaftlichen Rechnens befassten?

DK: Mein anfänglicher Schwerpunkt lag auf Programmiersprachen; ich wurde Redakteur dieser Sektion in CACM, dann JACM. Während ich mein Buch schrieb, lernte ich bald, dass Compiler-Autoren auch Techniken von allgemeinem Interesse entwickelten, und dass es großen Spaß machte, diese Algorithmen quantitativ zu analysieren.



Abb. 7: Marshall Hall Jr (John Simon Guggenheim Memorial Foundation)

Auf der ICM 1970 (Nizza, Frankreich) war schon der Titel Ihres Vortrags »*The Analysis of Algorithms*« ein Novum. Welche entscheidende Idee wollten Sie vermitteln, auch durch die Wahl dieses neuartigen Titels? Aus historischer Sicht ist es wichtig zu wissen, dass die Informatik noch zu Beginn der 70er Jahre nur an einigen wenigen Universitäten und in einigen Avantgarde-Unternehmen auf großes Interesse stieß.

DK: Ende der 1960er Jahre war die Informatik dreigeteilt: Numerische Analyse, Programmiersprachen, künstliche Intelligenz. Aber keiner dieser Titel entsprach ganz meinen Interessen. Also erfand ich den Namen *Analysis of Algorithms*, dessen erste Definition lautete: Der Teil der Informatik, der mir am besten gefällt.

In den späten 60er Jahren verließen Sie Caltech. Kaum 30 Jahre alt, gingen Sie als ordentlicher Professor für Informatik nach Stanford. Erleichtert wurde dieser Schritt durch George Forsythe, der sowohl in Stanford als auch in der Welt der Informatik eine wohlwollende und beherrschende Stellung einnahm und den Sie sogar mit einem »Martin Luther der Computerreformation« verglichen haben; auch

Ihre Gedanken zu einer tiefen Freundschaft mit einem anderen Stanford-Kollegen, Robert W. Floyd.

DK: Floyd und ich waren Freunde, seit wir uns 1962 kennenlernten. Wir hatten einen spannenden Briefwechsel über Bose-Nelson-Sortiernetze. Wir wollten irgendwann an der gleichen Universität sein. Überall, außer in Stanford, müssten wir helfen, eine führende Abteilung aufzubauen; Forsythe hatte das bereits wunderbar gemacht.

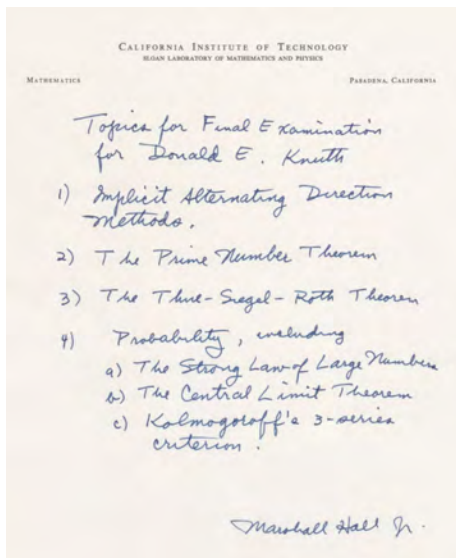


Abb. 8: Einen Monat vor seiner mündlichen Abschlussprüfung stellte M. Hall ihm vier Themen zur Auswahl und Don entschied sich für das dritte, über das er referieren sollte (1963 Handschriftliches Dokument, Donald Knuth).

matikerkollegen waren Dantzig in OR, sowie Leute wie Berlekamp, Gale, Karp, Lehmer in Berkeley.

Paul Cohen, ein älterer Kollege in Stanford, interessierte sich anfangs für Analysis, bevor ein angenehmes, schicksalhafteres Gespräch nach dem Abendessen mit einem Logiker-Kollegen Sol Feferman in einem gemeinsamen Stanford-Berkeley-Kolloquium sein Interesse an Kurt Gödels Ideen zur Kontinuumschypothese weckte. Dieses zufällige Ereignis spornte Cohen offenbar zu bedeutenden Fortschritten in der Logik an, die ihm schließlich sogar eine Fields-Medaille einbrachten, die 1966 verliehen wurde. Haben auch Sie das Berkeley-Stanford-Kolloquium von damals als anregend empfunden?

Sie haben 1969 in Stanford angefangen und sind seitdem nie weggegangen! Die Informatik in Stanford wurde aus dem Fachbereich Mathematik ausgegliedert, und letzterer hatte einige brillante Mathematiker auf seiner Liste. Da war der majestätische und staatsmännische Kunihiko Kodaira und sein häufiger Mitarbeiter Don Spencer. Da war das ungarische Duo Gábor Szegő und George Pólya, die beide sowohl in der Pädagogik als auch in der Forschung außergewöhnlich waren. Charles Loewner, Paul Garabedian, Paul Cohen und Solomon Feferman gehören ebenfalls zu dieser Ära. Ihre Gedanken zu Ihren berühmten Kollegen.

DK: Stanford hatte großartige Mathematiker, aber fast alle in der Analysis mit einem Algebraisten als Alibi. Cohen war brillant, hielt aber die Kombinatorik für trivial. Pólya war eine wunderbare Ausnahme; meine anderen Mathe-



Abb. 9: (v.l.n.r.) Die Knuths und die Carters (Jills Eltern – James und Wilda) bei seiner Hochzeit (1961 Kent Quisel).



Abb. 10: Don tippt seine Doktorarbeit auf einer neuen IBM Selectric-Schreibmaschine (1963 Jill Knuth).

DK: Cohen war auch am Caltech legendär. Aber ich habe eigentlich nie von einem Berkeley-Stanford-Mathe-Kolloquium gehört; bei Mathe-Vorträgen habe ich mich oft gefragt: »Na und, na und?« Ich begann eine Reihe wöchentlicher Mathe-Seminare zur Kombinatorik bei mir zu Hause, und oft nahmen Leute aus Berkeley daran teil.

Bei einem Treffen in Bukarest im Jahr 1971 sagten Sie: »Der Text meiner heutigen Predigt stammt von Platon (Die Republik, VII), der sagte: ›Ich habe kaum je einen Mathematiker gekannt, der vernünftig war‹. Wenn wir eine unvoreingenommene Prüfung der Leistungen der Mathematiker in der realen Welt der Computerprogrammierung vornehmen, müssen wir zu dem Schluss kommen, dass die Theorie bisher mehr Schaden als Nutzen angerichtet hat. Es gibt zahlreiche Fälle, in denen theoretische ›Fortschritte‹ die Entwicklung der Computerprogrammierung tatsächlich gebremst haben, und in einigen Fällen haben sie sogar dazu geführt, dass diese mehrere Schritte zurückging. Ich werde in diesem Vortrag auf einige solcher Fälle eingehen.« Ihre Gedanken zu Theorie und Praxis?

DK: Eine Woche vor dem Bukarester Kongress habe ich auf dem IFIP-Kongress in Ljubljana über die Schönheiten der Theorie gesprochen. So konnte ich mich auf



Abb. 11: Don und Jill am Caltech, kurz nach der Verleihung seines Doktorgrades (Juni 1963 Ervin Knuth).

beiden Seiten des Zauns bewegen. Ich hatte beide Extreme gesehen, als ich die Bände 1, 2 und 3 von *The Art of Computer Programming* schrieb. Überall finden wir Yin und Yang.

Ihre zehnjährige ästhetische Pilgerreise in den 70er Jahren, auf der Sie die Grundlagen des Druck- und Verlagswesens radikal umgestalteten, läutete eine völlig neue Ära für den Schriftsatz ein. 1979 erschien im *Bulletin of the American Mathematical Society* ein Artikel von Ihnen mit dem Titel »*Mathematical Typography*«, den Sie übrigens George Pólya zu seinem 90. Geburtstag gewidmet haben und in dem Sie, was nicht überrascht, bibliografische Referenzen anführen, die bis zu Luca Pacioli aus Venedig im Jahr 1509 zurückreichen. Wie kam es überhaupt zu der Idee eines strengen Ansatzes für die Ästhetik des Schriftsatzes?

DK: Ich liebe Bücher und Alphabete seit meiner Kindheit. Ich liebte das Aussehen der Texte von Addison-Wesley. Ich habe viele Jahre als Lektor, Zeitschriftenredakteur und Korrektor gearbeitet. Die Informatik brauchte neue typografische Tricks, die ich mitgestaltet habe. Viele gute Druckereien waren in der Nähe.

In Ihrer ICM-Rede von 1970 sagten Sie: »Jeder vernünftige Algorithmus wirft interessante Fragen ›rein mathematischer‹ Natur auf; und die Antworten auf diese Fragen führen manchmal zu nützlichen Anwendungen, was dem Thema ein wenig Schwung verleiht, ohne seine Schönheit zu verderben«. Wie haben Sie am Beispiel des Buchstabens »S« das Problem geometrisch formuliert, die »schönste Kurve« zu finden, die das »S« repräsentiert, und so die Mathematik mit der kreativen Ästhetik zu



verbinden, auch wenn Sie noch in einem rein algorithmischen Kontext arbeiten?

DK: Der Buchstabe »S« suggeriert ein schönes geometrisches Problem der Anpassung einer Ellipse an eine Tangente, das mit klassischen Primitiven lösbar ist. Das Anpassen von Kurven an Raster, das Zerlegen von Absätzen in Linien und viele andere reizvolle mathematische Herausforderungen ergeben sich auf natürliche Weise, die ich in meinem Vortrag in Epidaurus 1985 zusammengefasst habe.

Die mit dem Nobelpreis ausgezeichneten Physiker Paul Dirac und S. Chandrasekhar haben beide die Schönheit und ihre enge Verwandtschaft mit der Wahrheit auf ihre ganz eigene Weise gepriesen. Der unnachahmliche Computerwissenschaftler Edsger Dijkstra hatte ein berühmtes Zitat: »Ich finde Stil beim Programmieren wichtig«.

Abb. 12: Fahrradfahren mit Stoßdämpfern, erfunden von Alex Moulton (1965 Jill Knuth).

Ist, ganz im Sinne dieser drei Titanen, das unerbittliche und rigorose Streben nach Wahrheit ein bewährter Ansatz, um Werke von dauerhafter ästhetischer Schönheit zu schaffen, und zwar im gesamten Spektrum kreativer menschlicher Bemühungen?

DK: EWD316⁶ sagte auch, dass jeder »seinen eigenen Stil finden« sollte. Und ich glaube, jeder sollte seine eigene Schönheit finden. Aber seltsamerweise empfehle ich nicht, seine eigene Wahrheit zu finden! Für mich ist die Wahrheit völlig objektiv. Und ich bin froh, dass es Geheimnisse gibt, deren Wahrheit oder Falschheit man nicht kennen kann.



Abb. 13: In der Referenzbibliothek von Caltech, bei der Vorbereitung von Band 1 des TAOCP(1966 (Jill Knuth).

Die Kombinatorik und ihre verschiedenen Avatare haben in den letzten sechs Jahrzehnten ununterbrochen Ihre Aufmerksamkeit auf sich gezogen. Wenn sich einzelne Menschen grob in diejenigen einteilen lassen, die von Natur aus das Universum der diskreten Objekte (Graphen, Zahlen, Musiknoten, endliche Felder) schätzen, und diejenigen, denen das Universum der kontinuierlichen Objekte von Natur aus zusagt, wo positionieren Sie sich dann?

DK: Ich befinde mich eindeutig am diskreten Ende dieses Kontinuums. (Allerdings sind surreale Zahlen viel reicher als das Kontinuum selbst.) Wie Leibniz stelle ich mir gerne vor, dass alles aus 0en und 1en besteht. Wir nehmen Grenzen, wenn das eine nützliche Annäherung ist und wenn wir verstehen, was Grenzen bedeuten.

Wenn wir an diskrete Objekte und ihre Erforschung denken, besonders im 20. Jahrhundert, ist es fast unvermeidlich, an Paul Erdős zu denken. Sie und er hatten viele gemeinsame Interessen, und beide waren übrigens auch eingeladene Redner auf der ICM 1970 (Nizza, Frankreich). Ihre Erdős-Nummer ist allerdings 2 und nicht 1, wie die meisten Leute erwarten würden, Augen zu und durch. Erzähl uns eine gute Erdős-Geschichte!

DK: Als ich ihn in Nizza traf, erwähnte ich stolz, dass ich die überraschende Formel $l(382)=l(191)$ in der Theorie der Additionsketten entdeckt hatte. Ohne mit der

⁶ »A Short Introduction to the Art of Programming« von Edsger W. Dijkstra; Taschenbuch, 97 Seiten, veröffentlicht im August 1971 von der *Technischen Hogeschool Eindhoven*.

Wimper zu zucken, fragte er sofort, ob es unendlich viele Fälle mit $l(2n)=l(n)$ gäbe. (Dieses Ergebnis wurde drei Jahre später von Thurber bewiesen.)



Abb. 14: Don mit Maurice Nivat in einem Pariser Café (Mai 1967 Jill Knuth).

Es gibt eine Geschichte, nach der Bill Gates eine große Spende an die Informatikabteilung von Stanford machen wollte. Es handelt sich um denselben Bill Gates, an den sich Christos Papadimitriou erinnert, der sich kaum für die Forschung interessierte, obwohl ihre gemeinsame Arbeit zur Veröffentlichung in einer mathematischen Zeitschrift angenommen worden war. Bitte erzählen Sie uns die Geschichte hinter dieser Spende.

DK: Ein paar Stunden vor Bills Besuch hatte ich darüber nachgedacht, wie sich Zufallsgraphen bei ihrem Urknall entwickeln. Anhand der Übergangswahrscheinlichkeiten erkannte ich, dass der Nenner 17017 der Schlüssel zu dem ganzen Muster war. Begeistert zeichnete ich das Bild an die Tafel; Bill war beeindruckt.

Im Jahr 2016 sprachen Sie bei der Paris *Kanellakis Memorial Lecture* über »*Hamiltonian Paths in Antiquity*«. Darin erwähnten Sie die Beiträge verschiedener indischer Mathematiker zur Welt der Kombinatorik. Sie gingen auch ausführlich darauf ein, wie die Sanskrit-Sprache verwendet wurde, um kombinatorische Wahrheiten auf kreative Weise zu vermitteln, meist in einem lyrischen Metrum, das die Menschen dazu verleitete, sich auf die kombinierten Freuden des Lesens, Zählens und Rechnens einzulassen. Ist das alte Indien für Sie ein zivilisatorischer Schauplatz, an dem sich viele Ihrer eigenen Interessen auf potenziell interessante und aufschlussreiche Weise treffen?

DK: Ja! Die kaschmirischen Dichter des neunten Jahrhunderts, die geniale »*citrakāvya*« schufen, gehören sicherlich zu den Pionieren der Kombinatorik. Das Studium der Sanskrit-Prosodie hat auch andere Bereiche der Mathematik vorangebracht. Aber nur wenige Gelehrte haben sich an sie erinnert, nachdem andere Ideen in Mode kamen.

Fühlen Sie sich persönlich dem indischen Ansatz in der Mathematik, wie er von den frühen indischen Pionieren vertreten wurde, nahe? Die indische Herangehensweise war pragmatisch und konzentrierte sich auf die Lösung konkreter Probleme, wobei sie sich bewusst von einer Herangehensweise distanzierte, die zunächst eine Theorie oder sogar eine Axiomatisierung des Themas anstrebte. Die Manifestationen

dieser indischen Denkschule durchdringen die Werke von Āryabhaṭa, Bhāskara, Brahmagupta und anderen.

DK: Ich stimme nicht zu, dass die indische Herangehensweise völlig pragmatisch und nicht von Neugier getrieben ist. Ich fühlte mich sofort mit Nārāyaṇa verwandt, als ich von seinem Gaṇita Kaumudī (1356) erfuhr, denn ich hatte 1961 mit genau denselben Konzepten gespielt! Er muss sich im Jahr 1356 einsam gefühlt haben.



Abb. 15: Don, mit seinen Babys, beim Zelten mit Nivat in der Anza-Borrego-Wüste (März 1968 Jill Knuth).

Sie haben zugegeben, dass Sie eine enge Verwandtschaft mit Sprachen haben, insbesondere mit deren innerer Struktur. Seit Ihrer Schulzeit, als Sie in das Diagrammieren von Sätzen eingeführt wurden, bis zu der Zeit, als Sie während Ihrer Flitterwochen die Syntaktischen Strukturen von Noam Chomsky studierten, und sogar bis heute haben Sprachen Sie fasziniert. Könnten Sie bitte die Arbeit des indischen Grammatikers Pāṇini beleuchten und erläutern, inwiefern seine Arbeit, die vor vielen Jahrtausenden geleistet wurde, für das heutige Studium und die Verwendung von Computersprachen von Bedeutung ist?

DK: Mathematik ist die Wissenschaft der Muster. Sprachen sind vielleicht die komplexesten

Artefakte der Zivilisation. Pāṇini hatte die tiefe Einsicht, dass Sanskrit Muster enthält, die formalisiert und in ein logisches System gebracht werden können. Er gab seinen Kommentatoren über Jahrtausende hinweg reichlich Stoff zum Nachdenken.

Wenn man versucht, die beabsichtigte semantische Interpretation eines gegebenen Satzes aus der Sicht der auf Sanskrit angewandten natürlichen Sprachverarbeitung (*Natural Language Processing, NLP*) algorithmisch zu entschlüsseln, steht man unmittelbar vor der Herausforderung, dass es eine Vielzahl möglicher Bedeutungen gibt, die sich alle erheblich voneinander unterscheiden, was auf die Feinheiten der Worttrennungen im Sanskrit zurückzuführen ist. Selbst eine geringfügig abweichende Teilung eines zusammengesetzten Wortes kann zu einer völlig anderen semantischen Interpretation führen. Wie kann man solche unerwünschten Situationen algorithmisch zähmen oder eindämmen?

DK: Ich kenne das Sanskrit nur aus dritter Hand, aber man hat mir gesagt, dass es genau diese Mehrdeutigkeit ist, die Dinge wie citrakāvya möglich und inspirierend gemacht hat. Andererseits scheinen moderne NLP- und ML-Methoden die Komplexität beherrschbar gemacht zu haben, wenn erst einmal genügend Daten gesammelt wurden.



Abb. 16: Don spielt mit Sohn John, an Weihnachten (1969 Jill Knuth).



Abb. 17: Mit Tochter Jennifer in Stanford, am selben Klavier spielend wie in Milwaukee (1973 Jill Knuth).

Das obige Beispiel inspiriert uns dazu, den Bereich der sehr großen Zahlen heranzuziehen. Ihr guter Freund, der verstorbene Ronald Graham, wird mit einer gleichnamigen Zahl in Verbindung gebracht, die auch zu den größten Zahlen gehört, die jemals in einem veröffentlichten Beweis einer mathematischen Aussage verwendet wurden. Ist die Idee einer »größten positiven Zahl« nicht ein Oxymoron, da es immer eine Zahl gibt, die nur um eins größer ist als sie selbst?

DK: Nein, die größte positive Zahl, die in einem veröffentlichten Beweis verwendet wird, ist kein Oxymoron (und auch keine Konstante)! In meinem Vortrag über den Umgang mit der Endlichkeit wurde $Super - K = 10 \uparrow \uparrow \uparrow 3$, was viel größer ist als Rons Zahl; ich denke, sie ist zu groß, um wirklich verstanden zu werden.

Wie in der Nature-Ausgabe vom 4. Februar 2021 berichtet, hat ein Team israelischer Informatiker Ideen aus der Gradientenabstiegs-optimierung genutzt, um mathematische Ausdrücke für fundamentale Konstanten aus der Natur wie »e« und » π «. Sie nannten ihren Algorithmus die Ramanujan-Maschine und fanden sogar neuartige Ausdrücke für die katalanische Konstante, ein in der Kombinatorik allgegenwärtiges Objekt. Das Thema der kombinatorischen Identitäten selbst ist ein altes und gut erforschtes Thema, und Sie haben sogar ein Vorwort für den Klassiker von Marko

Petkovsek, Herb Wilf und Doron Zeilberger geschrieben, der selbst den schrulligen Titel $A = B$ trägt.

DK: In der Tat stoße ich immer wieder auf scheinbar neue Muster, die nur sehr elementare Ideen beinhalten. So habe ich beispielsweise in Zeilbergers Seminar im Januar über die faszinierende Tchoukaillon-Anordnung gesprochen, die jede positive ganze Zahl genau einmal enthält; ihre Eigenschaften sind weitgehend unerforscht.



Abb. 18: Die ACM Turing-Vorlesung in San Diego, Kalifornien (11.11.1974 Jill Knuth).

In der Ausgabe der Zeitschrift *Nature* vom 10. Juni 2021 erhielten Forscher von *Google Brain*, die an einem besonders heiklen, jahrzehntealten Problem des VLSI-Designs, dem so genannten *Chip Floorplanning*, arbeiten, einen Durchbruch, der den sonst mühsamen und teuren Aufwand eines Teams menschlicher Experten, das monatelang arbeitet, auf nur sechs Stunden reduziert. *Deep Reinforcement Learning*, das Lernverfahren, das diesem Fortschritt zugrunde liegt, wird sowohl von der Industrie als auch von der Wissenschaft als Durchbruch gefeiert und ist ideal für Probleme mit kombinatorischen Daten geeignet. Können die Leser in Anbetracht der zentralen Bedeutung kombinatorischer Ideen für Ihren Ansatz in den kommenden *TAOCP*-Ausgaben relevantes Material zu KI und ML erwarten?

DK: Nein; KI und ML sind Themen, über die andere viel besser schreiben können als ich.

Natürlich gibt es viele Arten des Lernens und viele Arten der Kombinatorik. *TAOCP* diskutiert zum Beispiel die Verstärkung in Message-Passing-Modellen von zufälligen Erfüllbarkeitsproblemen. Aber das ist etwas anderes.

Als David Mumford in der Januar-Ausgabe 2021 von Bhāvanā gefragt wurde, was er von Deep Learning halte, sagte er, dass er anfangs skeptisch war, bis er auf verwandte Arbeiten von Chris Manning stieß. Wir zitieren Mumford wortwörtlich: »Manning stellte diese beiden Fragen: Wie funktionieren diese Deep-Learning-Algorithmen zur Analyse von Sätzen? Entdecken sie die Grammatik?«. Mumford führt weiter aus, dass er schließlich davon überzeugt war, dass Mannings Schaffung einer geeigneten hochdimensionalen Darstellung von Wörtern bereits implizit die bis dahin schwer fassbare Grammatik des Satzes in sich trug. Werden wir erleben, dass Maschinen den Menschen nicht nur bei mechanischen Aufgaben, sondern auch bei intellektuellen Aufgaben voraus sind?

DK: Ich bin immer noch auf der Seite von Mumford, wenn es darum geht, nicht zu glauben, dass ML diskrete Dinge wie orthogonale lateinische Quadrate finden kann (die ich und mein Mentor R. C. Bose geliebt haben). Das große potenzielle Problem ist, dass niemand in der Lage ist zu verstehen, wie tiefe neuronale Netze zu ihren Schlussfolgerungen kommen.

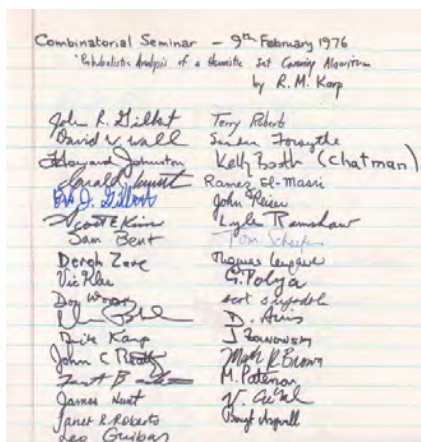


Abb. 19: Gäste der Seminare trugen sich in ein spezielles Protokollbuch ein, in dem alle relevanten Details festgehalten wurden (Februar 1976, Handschriftliches Dokument, Donald Knuth).



Abb. 20: Jimmy Carter verleiht Don die Nationale Wissenschaftsmedaille im Weißen Haus (14.1.1980 Jill Knuth).

Was die Frage $P = NP$ angeht, so haben Sie früher die Position vertreten, dass P nicht gleich NP ist. Aber wir haben gehört, dass Sie jetzt einen anderen Standpunkt vertreten. Wurde diese Änderung durch Laszlo Babais Behandlung des Graphen-Isomorphismus-Problems ausgelöst? Oder hat es mit dem Quantenalgorithmus von Peter Shor zu tun, der es erfolgreich mit dem klassischen RSA-Algorithmus aufgenommen hat?

DK: Nein. Ich vermute, dass $P = NP$ ist, weil ein Polytime-Algorithmus existieren könnte, ohne verständlich zu sein (sogar noch mehr als Super-K). Existenz ist etwas ganz anderes als Verkörperung. Robertson und Seymour haben gezeigt, dass es für einige Graphenprobleme Polytime-Algorithmen gibt, die aber wahrscheinlich nicht bekannt sind.

Der große Israel M. Gelfand nannte Schönheit, Exaktheit, Einfachheit und die Verücktheit von Ideen als das, was die Mathematik im Wesentlichen mit Musik und Poesie gemeinsam hat. Da Sie zu gleichen Teilen Wissenschaftler und Künstler sind, wie Ihre einzigartige Bezeichnung in Stanford beweist, glauben Sie, dass die vier hier aufgeführten Ideen notwendig und ausreichend sind, um Ideen von bleibendem ästhetischem Wert zu schaffen? Was ist mit Leidenschaft?

DK: Hmm. Warum hat Gelfand den Begriff »Spaß« weggelassen? Außerdem lehnt die japanische Philosophie des Wabi-Sabi ausdrücklich ab, dass Exaktheit notwendig oder ausreichend ist. Meine musikalischen und poetischen Freunde sind auch keine

Fans von Exaktheit; perfekter Rhythmus ist zu langweilig. Leidenschaft und je-ne-sais-quoi⁷ sind unverzichtbar.

Die aktive Teilnahme an der lutherischen Kirche und die leidenschaftliche Beschäftigung mit der Musik haben einen großen Teil Ihres Lebens außerhalb des akademischen Bereichs ausgemacht. Wie hat die Musik selbst Ihre Wertschätzung für die Kombinatorik beeinflusst, oder ist es umgekehrt? Sehen Sie in diesem Zusammenhang Parallelen zwischen Musik und Kombinatorik, insbesondere in den Werken der Meister?

DK: Jeder, der sich beispielsweise die Partituren von Tschaikowsky ansieht, erkennt, dass er ein großer Kombinatoriker war. Es gibt starke historische Bezüge (Werke von Śārngadeva, Mersenne und Schillinger explizit, Bach implizit). Diese Woche genieße ich die fantastische Kombinatorik von Burt Bacharach.

Von *Fairchild Semiconductors* über *SUN Microsystems* bis hin zu Google – eine ganze Reihe äußerst erfolgreicher und profitabler Technologieunternehmen sind allesamt irgendwann einmal aus Stanford hervorgegangen. Haben Sie persönlich jemals davon geträumt, Unternehmer zu werden, wenn man bedenkt, dass Sie der weltweit führende Guru der Programmierung sind und viele Fortschritte beaufsichtigt und geleitet haben, die den ideologischen Treibstoff für die Revolution geliefert haben, die buchstäblich vor Ihrem Fenster im Silicon Valley stattfindet?

DK: Ich verstehe, warum meine Freunde Erfüllung finden, wenn sie andere beaufsichtigen oder den Markt beeinflussen. Aber das Profitmotiv war immer am weitesten von meinen Gedanken entfernt, nachdem ich einen festen Job hatte. Ich beneide die Astronomen, weil die Leute verstehen, dass Astronomen durch Neugierde motiviert sind.

Die Geschichte der Kombinatorik ist reich und voll von erstaunlichen Persönlichkeiten und Errungenschaften. Da Sie es lieben, Ideen in einen angemessenen historischen Rahmen zu stellen, wie die Anspielung auf ein Manuskript aus dem Jahr 1509 n. Chr. in der Bibliographie Ihres Artikels über Typografie im Bulletin of the AMS zeigt, würden wir gerne wissen, wer Ihre eigenen Helden des kombinatorischen Denkens sind. Da diese Frage aufgrund der schieren Altertümlichkeit des Themas möglicherweise nicht ganz einfach zu beantworten ist, wollen wir uns auf die letzten sechs Jahrhunderte konzentrieren, und zwar in linearer Reihenfolge, beginnend mit dem 16.

DK: Gute Frage! Girolamo Cardano (1501-1576); John Wallis (1616-1703); Leonhard Euler (1707-1783); James Joseph Sylvester (1814-1897); Jack Edmonds (1934-); László

⁷ Die Bedeutung des französischen Ausdrucks »Je-ne-sais-quoi« bezieht sich auf etwas (z. B. eine ansprechende Eigenschaft), das nicht angemessen beschrieben oder ausgedrückt werden kann.

Lovász (1948-). Und lassen Sie mich noch Richard Stanley (1944-) hinzufügen, ein Student in der ersten Kalkulationsklasse, die ich unterrichtete (1963).

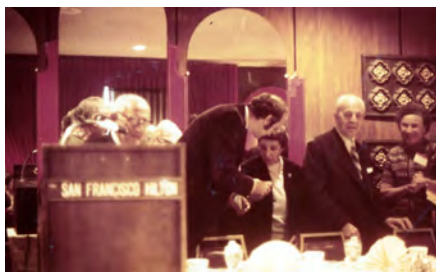


Abb. 21: (v.l.n.r.) George Pólya, Don, Anna Szegő, Gabor Szegő, und Emma Lehmer. Bei einem Mittagessen zur Feier des 50. Jahrestages von Pólya und Szegő's klassischem Werk *Aufgaben und Lehrsätze aus der Analysis* (Januar 1974 Jill Knuth).



Abb. 22: (v.l.n.r.) Victor Klee, George Pólya, Redner Richard Karp und Don in seinem Haus in Stanford, wo er in den 70er Jahren Seminare über Kombinatorik abhielt (Jill Knuth).

Sie sind weithin gefeiert worden und haben den *Turing Award*, den *Kyoto-Preis*, die *National Medal of Science*, die *John von Neumann Medal* und den allerersten *ACM Grace Murray Hopper Award* gewonnen. Es gibt sogar einen Asteroiden im Weltraum mit dem Namen *21656 Knuth*. Aber überraschenderweise halten Sie sich von E-Mail und elektronischer Kommunikation fern und ziehen es vor, nur noch Sekretariatskräfte zu beschäftigen, die sich durch Säcke voller Schneckenpost wühlen. Wie schaffen Sie es, Ihren vollen Terminkalender zwischen den zukünftigen Ausgaben des *TAOCP* und anderen akademischen Verpflichtungen wie der Rolle des Orakels einer dankbaren und anerkennenden Gemeinschaft von Informatikern und Programmierern zu bewältigen?

DK: Ich habe keinen Fernseher. Ich bin seit 1993 im Ruhestand. Ich arbeite im »Batch-Modus«, nicht im »Swap-in-swap-out«. Ich schreibe und überarbeite *TAOCP* ein Wort nach dem anderen. Ich schwimme fast jeden Tag. Mache häufig Nickerchen. Ich spiele Klavier und Orgel. Besuche das Stanford-Theater. Habe eine liebevolle Familie. Habe zahllose Helfer. Schokolade.

Richard Hamming sagte: »Der Zweck des Rechnens ist Erkenntnis, nicht Zahlen«. Um den Kontext zu verdeutlichen, hier ein Zitat des verstorbenen Vladimir Voevodsky (*Fields-Medaille*, 2002), der sich Computer als Assistenten für professionelle Mathematiker vorstellte, die ihnen helfen, die Korrektheit der von ihnen erstellten Beweise automatisch zu überprüfen. »Die Welt der Mathematik wird sehr groß, die Komplexität der Mathematik wird sehr hoch, und es besteht die Gefahr, dass

sich die Fehler häufen. Beweise stützen sich auf andere Beweise; wenn einer einen Fehler enthält, werden alle anderen, die sich auf ihn stützen, den Fehler teilen.« Ist dies der Vorbote einer synergetischen intellektuellen Beziehung zwischen Mensch und Maschine?



Abb. 23: (v.l.n.r.) Don, Oren Patashnik, Ron Graham und Fan Chung bei einer Signierstunde zur Einweihung der Veröffentlichung von *Concrete Mathematics* (1988 Jill Knuth).



Abb. 24: (von links nach rechts) Mario Capecchi, Donald Knuth und Willard V. O. Quine; Kyoto-Preisträger von 1996 (Jill Knuth).

DK: (Außerdem sollte das Ziel des maschinellen Lernens Einsicht sein, nicht Ergebnisse!) Es ist wunderbar, dass Computer die menschlichen Fähigkeiten ergänzen, nicht nur bei der Überprüfung von Beweisen, sondern auch bei symbolischen Berechnungen usw. Natürlich können Computerprogramme Fehler haben; das gilt auch für ihre Gültigkeitsnachweise.

Der Humor in Uniform hat Sie offenbar all die Jahre bei guter Laune gehalten! In einem Kurs mit dem Titel »Konkrete Mathematik«, den Sie abteilungsübergreifend in Stanford anboten und in dem Sie aus Ihrem gleichnamigen Buch lehrten, hofften die Studenten, dass sie etwas »harte« und nicht »weiche« Mathematik lernen würden. Als Sie in der Vorlesung sagten, dass es keine »Theorie der Aggregate«, kein »Stone's Embedding Theorem« und nicht einmal die »Stone-Cech Compactification« geben würde, verließ ein Haufen enttäuschter Bauingenieure still und leise den Raum.

DK: Die »Graffiti« am Rande von *Konkrete Mathematik*, die hauptsächlich von Studenten aus Princeton und Stanford beigesteuert wurden, als Ron und ich zum ersten Mal anhand von Vorabdrucken des Buches unterrichteten, waren eindeutig erfolgreich: Alle Fehler, die sie

enthielten, wurden von den Lesern vor allen Fehlern im Text entdeckt.

Wir verstehen, dass »Vorhersagen schwierig sind, besonders was die Zukunft betrifft«. Wir haben auch erfahren, dass Sie nicht allzu gern Prophezeiungen machen. Dennoch möchten wir Ihre Meinung zu Entwicklungen hören, die den Kern der Informatik mit Mathematik und Physik wie nie zuvor verschweißen. Was sollte der Don Knuth der Ära des maschinellen Quantenlernens im 21. Jahrhundert, z. B. ein Teenager, der irgendwo im Landesinneren Indiens ein Programm auf einer öffentli-

chen Quanten-Cloud laufen lässt, beachten, während er/sie sich auf die kommenden Herausforderungen und Möglichkeiten vorbereitet?

DK: Ich habe erfolglos versucht, Quantencomputing zu verstehen. Vielleicht gibt es Leute, die es verstehen, aber die Art von Computing, die ich betreibe, nicht nachvollziehen können. Ich weiß nur, dass es zwei völlig verschiedene Dinge zu geben scheint, die beide »Computing« heißen. Jeder sollte seinem eigenen Stern folgen.

Ich danke Ihnen für die anregenden Fragen. Bitte interviewen Sie weiterhin andere.

Lieber Professor Donald Knuth, dieses Gespräch war für uns eine inspirierende Erfahrung, auch wenn wir mit einem großen Pionier auf den Spuren der Vergangenheit wandelten. Wir wünschen Ihnen noch viele Jahre bei bester Gesundheit und weiterhin viel Erfolg in der Forschung und in anderen Bereichen des Lebens. Wir danken Ihnen!

Die Autoren

Sudhir Rao ist Gründungsmitglied und mitwirkender Redakteur von Bhāvanā. Er forscht in bestimmten Bereichen der Physik und Mathematik.

Indranath Sengupta ist Mitglied des Lehrkörpers der Abteilung für Mathematik am IIT Gandhinagar. Er ist korrespondierender Redakteur für Bhāvanā und Teammitglied des HoMI-Projekts des IIT Gandhinagar.

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 oder als RSS-Feed auf <https://ctan.org/> verfolgt werden.

oststud von *Naoki Pross* ist ein Paket für Dokumente der Studierendenorganisation der Fachhochschule Ostschweiz.

CTAN:macros/latex/contrib/oststud

drawing-with-metapost von *Toby Thurston* enthält mehr als 200 Beispiele für technische Illustrationen und Diagrammen samt Quellcode, die mit METAPOST erstellt wurden. Das Projekt dient zur Demonstration dessen, was möglich ist, es

vermittelt Anregungen.

CTAN:info/drawing-with-metapost

quickreaction von *Francesco Lambertini* stellt eine Umgebung und einen Befehl bereit, mit dem man chemische Reaktionsgleichungen und Reaktionschemata leichter auszeichnen und mit *TikZ* zeichnen kann.

CTAN:graphics/pgf/contrib/quickreaction

cleveref-usedon von *Sven Pistré* setzt auf dem Paket *cleveref* von *Toby Cubitt* auf und erweitert es um die Möglichkeit, mit der Floskel »used on pp. ...« eine Bezugnahme auf andere Stellen im Text herzustellen.

CTAN:macros/latex/contrib/cleveref-usedon

elteiktdk von *Máté Cserép* ist eine Klasse für Arbeiten zur ungarischen *National Conference of Scientific Students' Associations (OTDK)* und setzt dabei das Layout der Eötvös-Loránd-Universität Budapest um.

CTAN:macros/latex/contrib/elteiktdk

gelasio von *Bob Tennent* enthält die Schriftart *Gelasio* von *Eben Sorkin* samt der dazugehörigen $\text{\LaTeX}$ -Unterstützung. Die mageren und fetten Schnitte der Schrift sind kompatibel mit der Georgia. Mittel- und halbfette Schnitte wurden ergänzt.

CTAN:fonts/gelasio

osda von *Christian Krieg* ist ein Paket für Veröffentlichungen in den *Proceedings of the Workshop on Open-Source Design Automation*.

CTAN:macros/latex/contrib/osda

lua-tinyyaml von *Zeping Lee* ist ein Parser für YAML (subset), der in Lua geschrieben wurde. Es kann mit anderen $\text{\LaTeX}$ -Paketen verwendet werden.

CTAN:macros/luatex/generic/lua-tinyyaml

texfindpkg von *Jianrui Lyu* ist ein Lua-Skript, mit dem man die Abhängigkeiten von $\text{\LaTeX}$ -Paketen, -Befehlen und -Umgebungen ermitteln und gegebenenfalls nachinstallieren kann. Unterstützt werden $\text{\TeX}$ Live und $\text{\TeX}$ Live.

CTAN:support/findpkg

tikz-nfold von *Jonathan Schulz* zeichnet drei-, vier- und allgemein n-fache Pfade mit *TikZ*. Es ist kompatibel mit *tikz-cd* von *Augusto Stoffel*.

CTAN:graphics/pgf/contrib/tikz-nfold

latex-context-ppchtex von *David Carlisle* enthält ein Paket, mit dem man $\text{\picTeX}$ auch mit alten Dokumenten laden kann, die mit $\text{\TeX}$ ohne $\text{\eTeX}$ zu kompilieren sind. Heutzutage ist es obsolet, $\text{\picTeX}$ kann bei neuen Dokumenten direkt verwendet werden.

CTAN:obsolete/macros/latex/contrib/latex-context-ppchtex

korigamik von *Kushagra Lakhwani* ist eine Dokumentenklasse, die auf *article* aufsetzt und unter anderem flexiblere Möglichkeiten zum Gestalten von Titelseiten

bietet.

CTAN:macros/latex/contrib/korigamik

tangocolors von *Hilmar Preuße* setzt auf dem Paket *xcolor* auf und erlaubt es, Farben aus der Tango-Farbpalette von tango.freedesktop.org/ in L^AT_EX zu verwenden.

CTAN:macros/latex/contrib/tangocolors

mpchess von *Maxime Chupin* dient zum Zeichnen von Spielstellungen und Spielzügen beim Schach mit Hilfe von METAPOST. Das Design ist angelehnt an die Darstellungen auf der Website <https://lichess.org/>.

CTAN:graphics/metapost/contrib/macros/mpchess

egpeirce von *Jukka Nikulainen* ist ein Paket zum Zeichnen von »existential graphs«, einem System von Notationen und Schlussregeln für logische Aussagen nach *Charles S. Peirce*.

CTAN:graphics/egpeirce

emo von *Robert Grimm* ermöglicht es, farbige Emojis direkt in ein L^AT_EX-Dokument einzufügen, unabhängig von der verwendeten T_EX-Engine. Wenn es die Engine möglich macht, wird die Schrift Noto verwendet, die die Emojis enthält. Anderenfalls werden PDF-Grafiken eingebunden, die automatisch aus den SVG-Quellen von Noto erzeugt werden.

CTAN:graphics/emo

*fenetreca*s von *Cédric Pierquet* setzt die Ausgabe von Computeralgebrasystemen (CAS) wie Xcas und Geogebra mit Hilfe von TikZ in L^AT_EX um.

CTAN:graphics/pgf/contrib/fenetreca

alchemist von *Engelbert Buxbaum* dient dazu, alchemistische und astrologische Symbole mit Unicode zu verwenden. Dazu können verschiedene Fonts eingesetzt werden (GNU Unifont, Symbola, Quivira).

CTAN:fonts/alchemist

babel-lithuanian von *Sigitas Tolušis* beinhaltet die babel-Unterstützung für Litauisch.

CTAN:macros/latex/required/babel/contrib/lithuanian

outilsgeometkz von *Cédric Pierquet* erstellt mit Hilfe von TikZ Darstellungen von Bleistift, Geodreieck und Zirkel, die für geometrische Zeichnungen benötigt werden.

CTAN:graphics/pgf/contrib/outilsgeometkz

uvaletter von *Michele Piazza* ist eine inoffizielle Nachahmung des Briefkopfs der Universität von Amsterdam, der bis dahin nur für Microsoft Word verfügbar war.

CTAN:macros/latex/contrib/uvaletter

fistrum von *David Davó* ist ein Fork von *lipsum*, der einen alternativen Blindtext erzeugt. Die Vorlage dazu wurde von <https://www.chiquitoipsum.com/> übernommen.

CTAN:macros/latex/contrib/fistrum

recorder-fingering von *Alan Munn* hilft beim Erstellen von Griffdiagrammen für barocke Blockflöten.

CTAN:macros/latex/contrib/recorder-fingering

evangelion-jfm von *Jing Huang* sind Font-Metriken für chinesische und japanische Schriften, die den horizontalen und den vertikalen Satz mit typografischen Feinheiten unterstützen.

CTAN:macros/luatex/generic/evangelion-jfm

tidyres von *Yifan Liang* stellt Befehle bereit, die man zum Setzen des Lebenslaufs für eine Bewerbung verwenden kann.

CTAN:macros/latex/contrib/tidyres

lualinalg von *Chetan Shirore* und *Ajit Kumar* arbeitet mit Vektoren und Matrizen innerhalb eines L^AT_EX-Dokuments mit Lua[®]T_EX.

CTAN:macros/luatex/latex/lualinalg

sacsymb von *Hari Nyaupane* stellt Symbole bereit, die zur Darstellung der »Orchestrated objective reduction (Orch OR)«, einer Bewusstseinstheorie, verwendet werden.

CTAN:graphics/pgf/contrib/sacsymb

resolsysteme von *Cédric Pierquet* stellt Befehle bereit, die mit Hilfe von *xint* und *pyluatex* Berechnungen über lineare Systeme durchführen können.

CTAN:macros/latex/contrib/resolsysteme

physics2 von *Tingxuan Zhang* besteht aus Modulen, die das Setzen mathematischer Ausdrücke vereinfachen sollen.

CTAN:macros/latex/contrib/physics2

exam-lite von *Rok Kukovec* dient zum einfacheren Erstellen von Übungsblättern für Prüfungen.

CTAN:macros/latex/contrib/exam-lite

lparse von *Josef Friedrich* ist ein Lua-Modul, das Key-Value-Optionen parsen kann. Der Name ist von dem des Pakets *xparse* abgeleitet, wobei der Anfangsbuchstabe *l* darauf hinweisen soll, dass *lparse* nur mit Lua[®]T_EX zu verwenden ist.

CTAN:macros/luatex/generic/lparse

tikzviolinplots von *Pedro Callil-Soares* zeichnet Violindiagramme, die verwendet werden, um in statistischen Auswertungen Wahrscheinlichkeitsverteilungen darzustellen.

CTAN:graphics/pgf/contrib/tikzviolinplots

naive-ebnf von *Yegor Bugayenko* kann eine kontextfreie Grammatik mit Hilfe einer vereinfachten erweiterten Backus-Naur-Form ausgeben.

CTAN:macros/latex/contrib/naive-ebnf

tramlines von *Elijah Z. Granet* kann Überschriften in juristischen Schriftsätzen mit doppelten Über- und Unterstrichen versehen, wie es in Großbritannien und im Commonwealth üblich ist (sogenannte *tramlines*).

CTAN:macros/latex/contrib/tramlines

uol-physics-report von *Jan Eike Suchard* ist eine L^AT_EX-Klasse zum Schreiben von Laborberichten, die für Physik-Studierende der Universität Oldenburg entwickelt wurde.

CTAN:uol-physics-report

pixelarttikz von *Cédric Pierquet* definiert einen Befehl und eine Umgebung, um Pixel-Art-Grafiken mit TikZ zu zeichnen.

CTAN:graphics/pgf/contrib/pixelarttikz

tangramtikz von *Cédric Pierquet* dient zum Zeichnen von Tangram-Legespielen mit Hilfe von TikZ.

CTAN:graphics/pgf/contrib/tangramtikz

expkv-bundle von *Jonathan P. Spratte* ersetzt die früheren Pakete *expkv*, *expkv-cs*, *expkv-def* und *expkv-opt* (die im Übrigen aber auf CTAN verbleiben sollen). Es wird ein erweiterbares Key-Value-Interface bereitgestellt. Caveat beim Umstieg auf das neue Bundle: Ein paar kleinere Änderungen können sich gegebenenfalls nachteilig auf die Rückwärtskompatibilität auswirken.

CTAN:macros/generic/expkv-bundle

overarrows von *Julien Labbé* ist ein Paket, mit dem man Pfeile über oder unter mathematische Ausdrücke zeichnen kann.

CTAN:macros/latex/contrib/overarrows

Spielplan

13. 7. 2023 **65. Mitgliederversammlung von DANTE e.V.**
Hotel Collegium Leoninum
Noeggerathstraße 34
53111 Bonn
<https://www.dante.de/veranstaltungen/dante2023/>



14. 7. – 16. 7. 2023 **TUG 2023 – The 44th Annual Meeting of the T_EX Users Group**
Presentations covering the T_EX world Typography & typesetting, fonts & design, publishing and more
Hotel Collegium Leoninum
Noeggerathstraße 34
53111 Bonn
<https://www.tug.org/tug2023/>



28. 7. – 29. 7. 2023 **BayT_EX 2023**
Genua GmbH
Domagkstraße 7
85551 Kirchheim bei München
<https://www.ks-ingenieurconsult.de/TeX/Stammtisch.html>



10. 9. – 16. 9. 2023 **ConT_EXt Meeting 2023**
Prag-Sibřina, Tschechien
<https://meeting.contextgarden.net/2023/>



April 2024 (geplant) **DANTE 2024**
Wielandgut Oßmannstedt
Wielandstraße 16
99510 Ilmtal-Weinstraße, OT Oßmannstedt

Stammtische



In verschiedenen Städten im Einzugsbereich von DANTE e.V. finden regelmäßig Treffen von $\text{\TeX}$ -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

Mailingliste: [https://lists.rwth-aachen.de/postorius/lists/tex-stammtisch.](https://lists.rwth-aachen.de/postorius/lists/tex-stammtisch.lists.rwth-aachen.de)

lists.rwth-aachen.de

»Anvers«, Kockerellstr. 20, 52062 Aachen

Erster Donnerstag im Monat, 20:00 Uhr

Bad Doberan

Carsten Vogel

texnicer@web.de

zur Zeit inaktiv, Interessenten bitte per Mail melden

Berlin

Michael-E. Voges, Tel.: 0 33 62/ 50 18 35,

mevoges@t-online.de

»La Esperanza Restaurant Tapas Bar«, Chausseestr. 131B, 10115 Berlin

Zweiter Donnerstag im Monat, 19:00 Uhr

Darmstadt

Karlheinz Geyer

geyerk@posteo.de

ab 2023 Neustart geplant, Interessenten können sich bei Karlheinz melden

Erlangen

Peter Seitz

p.seitz@KplusS-Ing.de

<https://www.ks-ingenieurconsult.de/TeX/Stammtisch.html>

Gaststätte »Deutsches Haus«, Luitpoldstr. 25, 91052 Erlangen

Dritter Dienstag im Monat, 19:00 Uhr

Frankfurt a. Main

Harald Vajkonny

vajkonny@t-online.de

zur Zeit inaktiv, Interessenten bitte per Mail melden

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

z. Zt. inaktiv. Bei Fragen steht Günther gern per Mail zur Verfügung.

Hannover

Reiko Kaps

kaps@luis.uni-hannover.de

<http://tex-hannover.de/>

*RRZN/LUIS, 3D-Raum, Schloßwender Str. 6 (Gebäude 1210), 30159 Hannover
Zweiter Donnerstag im Monat, 18:30 Uhr, Terminabsprache über Mailingliste*



Heidelberg

Martin Wilhelm Leidig, Tel.: 01 70 41 83 32 9,

moss@moss.in-berlin.de

Anmeldeseite zur Mailingliste: <https://tinyurl.com/stammtisch-HD>

*letzter Freitag in Biergartenmonaten (ca. April bis Oktober), möglichst im Freien
Details und Abweichungen werden über die Mailingsliste bekannt gegeben.*



Köln

Uwe Ziegenhagen

uwe@dante.de

zur Zeit inaktiv, Interessenten bitte per Mail melden

Leipzig

Erhard Pross

Erhard.Pross@gmx.de

nächstes Treffen am 11.Mai 2023 18:00 Uhr

le-tex publishing services GmbH, Weissenfelder Str. 84, 04229 Leipzig

München

Leah Neukirchen

leah@vuxu.org

*Erste Woche in geraden Monaten, wechselnde Wochentage und Orte, 19:00 Uhr
Info folgt über Mailingliste dante-ev@dante.de*

Stuttgart

Bernd Raichle

bernd.raichle@gmx.de

Zweiter Dienstag im Monat, 19:30 Uhr

»Wichtel«, Bahnhofstr. 30, 70372 Stuttgart - Bad Canstatt

Adressen

DANTE, Deutschsprachige Anwendervereinigung T_EX e.V.

Postfach 11 03 61

69072 Heidelberg

Tel.: (0 62 21) 2 97 66

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:	Uwe Ziegenhagen	vice-president@dante.de
Schatzmeisterin:	Doris Behrendt	treasurer@dante.de
Schriftführer:	Volker RW Schaa	secretary@dante.de
Beisitzer:	Klaus Höppner	
	Harald König	
	Stephan Lukasczyk	

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

Luzia Dietsche 71394 Kernen dtkred@dante.de	[3]	Henning Hraban Ramm hraban@fieee.net	[57]
Willi Egger w.egger@boede.nl	[26,38]	Sudhir Rao Siehe Seite 90	[60]
Jürgen Fenn Neu-Isenburg juergen.fenn@gmx.de	[90]	Oliver Rath oliver_rath@genua.de	[6]
Ralf Mispelhorn Am Wettbach 12/2 72336 Balingen mispelsoft@mispelhorn.de	[19]	Indranath Sengupta Siehe Seite 90	[60]
Rolf Niepraschk Persiusstr. 12 10245 Berlin Rolf.Niepraschk@gmx.de	[25]	Martin Sievers siehe Seite 98	[4,5]
		Keno Wehr wehr@abgol.de	[8]

Die T_EXnische Komödie

35. Jahrgang Heft 2/2023 Mai 2023

Impressum

Editorial

Hinter der Bühne

- 4 Grußwort
- 5 Einladung zur 65. Mitgliederversammlung
- 6 Einladung zur BayT_EX 2023

Bretter, die die Welt bedeuten

- 8 L^AT_EX und Schulphysik 2: Schaltbilder
- 19 Musik-CD mit Cover produzieren
- 25 Mehrsprachigkeit: Das L^AT_EX-Paket translator
- 26 PocketDiary – ein Modul für Terminkalender
- 38 Andere Kalender mit dem PocketDiary-Modul
- 57 ConT_EXt kurz notiert

Von fremden Bühnen

- 60 Die Morgendämmerung der Strenge in der Kunst des Programmierens
- 90 Neue Pakete auf CTAN

Spielplan

- 95 Termine
- 96 Stammtische

Adressen

- 99 Autoren/Organisatoren