

Die T_EXnische Komödie

dante

Deutschsprachige
Anwendervereinigung T_EX e.V.

36. Jahrgang Heft 3/2024 August 2024

3/2024

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- $\text{\LaTeX}$ -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, $\text{\LaTeX}$ -Pakete oder Schriften notwendig, so müssen auch diese komplett mitgeliefert werden. Außerdem müssen sie auf Anfrage Interessierten zugänglich gemacht werden. Weitere Informationen für Autoren findet man auf der Projektseite <https://projekte.dante.de/DTK/AutorInfo> von DANTE e.V.

Diese Ausgabe wurde mit LuaHB $\text{\TeX}$, Version 1.18.0 (T $\text{\TeX}$ Live 2024) erstellt. Als Standard-schriften kamen Libertinus Serif, Libertinus Sans Serif, Anonymous Pro und Libertinus Math zum Einsatz.

Erscheinungsweise: vierteljährlich

Erscheinungsort: Heidelberg

Auflage: 2000

Herausgeber: DANTE, Deutschsprachige Anwendervereinigung T $\text{\TeX}$ e.V.

Bergheimer Straße 147

69115 Heidelberg

Neue Adresse!

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 Rolf Niepraschk Stefan Pinnow

Bernd Raichle Christine Römer HG Unckell

Herbert Voß

Redaktionsschluss für Heft 4/2024: 15. Oktober 2024

ISSN 1434-5897

Die T_EXnische Komödie 3/2024

Editorial

Liebe Leserinnen und Leser,

ich freue mich sehr, dass in dieser Ausgabe die Rubrik »Bretter, die die Welt bedeuten« gut gefüllt ist. Angefangen mit einem weiteren Teil der mich extrem beeindruckenden Reihe von Keno Wehr über $\LaTeX$ und Schulphysik.

Dem folgt der Bericht über den Einsatz von TikZ in einem nicht-naturwissenschaftlichen Bereich. Es beruhigt mich ungemein, dass ich nicht die Einzige bin, die immer wieder aufs Neue über die Einsatzmöglichkeiten von TikZ begeistert ist.

Die praktische Anwendung des Beamer-Stils Moloch, dessen Beschreibung folgt, konnte ich bereits bei einem Vortrag während der Tagung in Weimar in diesem Frühjahr sehen.

Auch Schriften kommen diesmal nicht zu kurz, ebenso wie die Neuigkeiten zu ConTeXt , die unter anderem von einem Handbuch zu »Mathematics in ConTeXt « künden. Ob sich da wohl jemand zu einer Besprechung in einer der nächsten Ausgaben bereit findet?

Ich selbst mische ebenfalls in der Rubrik mit, indem ich unter dem Titel »Aufgeschnappt« tidbits weitergebe, die ich in der großen, weiten Welt des Internets aufgeschnappt habe. Wer mag, kann mir gerne weitere solche Hinweise direkt liefern.

Ich wünsche auch diesmal viel Spaß beim Lesen und verbleibe mit $\text{T}_{\text{E}}\text{X}$ nischen Grüßen



Luzia Dietsche

PS: Bitte beachten – DANTE e.V. hat eine (mittlerweile nicht mehr ganz so) neue postalische Adresse (siehe Impressum).

Hinter der Bühne

Vereinsinternes

Grußwort

Liebe Mitglieder,

gerade erst ist die Fußball-EM vor unserer Haustür zu Ende gegangen, da ziehen auch schon die Olympischen Spiele in Paris die Aufmerksamkeit auf sich. Dazwischen der beginnende Ferien- und Urlaubsbetrieb – gar nicht so einfach, in dem Trubel des Alltags das Grußwort in dem gewohnten Umfang zu schreiben. Und dann war und ist da auch noch Corona, kurzum: Sehen Sie es mir bitte nach, dass ich mich diesmal etwas kürzer fasse.

Ende der vergangenen Woche (18.–21. Juli) fand die TUG 2024 in Prag statt – eine Veranstaltung, die ebenso wie die sportlichen Highlights dieses Jahres Teilnehmerinnen und Teilnehmer aus aller Welt anlockte, wenn auch in *etwas* kleinerer Zahl. Auch aus den Reihen von DANTE e.V. waren wieder viele Personen beteiligt.

Für uns als Verein steht im nächsten Jahr die Frühjahrstagung in Darmstadt auf dem Programm (3.–5. April 2025). Dazu stimmt sich der Vorstand aktuell mit dem lokalen Organisator Torsten-Karl Stempel ab. Er hat uns bereits 2019 an der Hochschule begrüßt und offenbar hat es auch ihm damals gefallen. Es wird sicherlich eine sehr schöne Veranstaltung mit einem interessanten Vortrags- und Begleitprogramm. Halten Sie sich also den Termin nach Möglichkeit frei.

Im »Büro« ist grundsätzlich eine gewisse Routine eingekehrt, doch steht uns mit der Umstellung auf VEWA ein größerer Arbeitsschritt bevor. Wir versuchen, gerade auch hier die anfallenden Arbeiten auf mehrere Schultern zu verteilen, doch ist insbesondere die Buchhaltung ein Bereich, in dem eine klare und einheitliche Linie nötig ist. Klar ist, dass auch diese kommerzielle und etablierte Software kein Selbstläufer ist, auch weil wir bei der Einrichtung abhängig vom Zeitplan der Firma sind. Wir hoffen aber weiterhin, möglichst bald ein stabiles System zu haben, um die Wirtschaftsjahre 2023 und 2024 damit ordentlich abzuschließen und 2025 dann von Beginn an routiniert arbeiten zu können.

Und nun viel Spaß mit der Lektüre dieser Ausgabe der Mitgliederzeitung.

Herzlichst Ihr/Euer

Martin Sievers

Bericht der Schatzmeisterin

Doris Behrendt

Stand Mitgliederversammlung, 06.04.2024

Wie in der Mitgliederversammlung in Weimar erläutert, habe ich es noch nicht geschafft, den Jahresabschluss für das Jahr 2023 zu erledigen. Einige Erläuterungen zu den Gründen finden Sie hier im Bericht.

Bankkonten

In Tabelle 1 sei € ein Platzhalter für das Buchungskonto der Barkasse. Die Kasse für die Jahre 2023 und 2024 ist noch nicht aufgearbeitet und somit der Kassenstand nicht bekannt, es handelt sich aber nur um geringe Posten im dreistelligen Bereich.

Kst.	Konto	Stand 01.01.2023	Stand 31.12.2023	Saldo 2023	Stand 31.03.2024
001	Kasse	9,29	€*	€	€
011	VR-Vereinsgirokonto	27.660,38	9.613,05	−18.047,33	5.956,61
012	VR-Cash	58.214,93	48.281,16	−9.933,77	48.305,08
020	VR-Mehrzinssparen	10.168,48	10.189,18	20,70	10.189,18
031	Paypal	558,94	752,49	193,55	802,15
041	Degussa	35.573,01	35.896,24	323,23	35.986,05
Summe		132.185,03	€ + 104.732,12	€ − 27.452,91	€ + 101.239,07

* € ist ein unbekannter Kassenstand zum 31.12.2023.

Tab. 1: Stand der Bankkonten

Einzüge

Im Jahr 2023 habe ich aufgrund des Wegfalls des Sekretariats die Mitgliedsbeiträge bei den Mitgliedern, die einen Bankeinzug wünschen, nicht durchgeführt. Es war mir nicht möglich sicherzustellen, dass die Bankdaten (IBAN etc.) in unserer Datenbank alle korrekt waren.

Zwar stand auf den Rechnungen, die den Mitgliedern zugestellt worden waren, eine Information hierzu, zusammen mit der Bitte um »heuer ausnahmsweise Selbstüberweisung«, aber dieser Bitte waren 671 Mitglieder nicht nachgekommen. Setzt man

im Mittel 40 € Mitgliedsbeitrag an, ergibt das 26.840 €, was recht genau dem Saldo von 2023 entspricht.

Tätigkeitsbericht

Aus Zeitgründen unterließ ich es, über meine Tätigkeiten zu berichten, denn die Mitgliederversammlung war bereits weit fortgeschritten.

Stand 15.07.2024

Einnahmen/Ausgaben 2023

Aufgrund der noch nicht erfolgten Buchungen für 2023 kann ich hierzu leider noch keine Zusammenfassung liefern.

Buchhaltung, Software

Wie an der Mitgliederversammlung kurz angerissen, hat der Anbieter *Campai* der Vereinsverwaltungssoftware seine Versprechen vom letzten Jahr nicht gehalten, insbesondere, was die Buchhaltungskomponente betrifft.

Deshalb wechseln wir nun wieder zu VEWA, was zwar wesentlich teurer ist, aber vorerst die beste, da schnellste Lösung. Wir müssen vorankommen und brauchen einen verlässlichen Partner.

Die Kosten für die Einführung/-richtung der Software werden sich einmalig auf ca. 13.000 € belaufen – vielleicht weniger, denn hier sind Schätzwerte enthalten für die Arbeitszeit, die beim Anbieter GRÜN anfallen. Dazu fallen nun monatlich ca. 700 € für die regelmäßige SAAS-Gebühr inklusive Support an.

Die Verträge sind unterschrieben, ein Treffen mit dem organisatorischen Projektleiter fand bereits statt. Die Software wird wahrscheinlich erst zu Beginn des Jahres 2025 mit allen Komponenten einsatzbereit sein.

Einzüge, Jahresrechnungen

Wir (Uwe und ich) arbeiten aktuell an den Bankeinzügen, allerdings ist sehr viel manuelle Zuarbeit nötig. Die Jahresrechnungen (auch für die Mitglieder ohne Bankeinzug) sind noch nicht erstellt. Ich stellte das zurück, da ich abwarten will, ob es bereits mit VEWA gemacht werden kann. Morgen wird es ein erstes Treffen mit dem Softwareprojektleiter unserer VEWA-Instanz geben, erst dann kann ich hier eine Entscheidung treffen. Da der Redaktionsschluss der Mitgliederzeitung aber schon heute ist, kann ich dazu noch nichts sagen.

Steuer, Freistellung

Im Bericht der Kassenprüfer in der „Die TēXnische Komödie“ 4/2023 auf Seite 25 f. wurde protokolliert, dass die Freistellung für 2018 bis 2020 noch nicht beantragt worden war. Dies ist falsch, denn der Freistellungsbescheid lag zu diesem Zeitpunkt bereits vor.

Die Frist zur Einreichung der Steuererklärung für die Jahre 2021 bis 2023 endet im August 2024. Ich werde sie nicht einhalten können, da zuerst die Buchhaltung, der Jahresabschluss und die Kassenprüfung für 2023 abgeschlossen sein müssen, ebenso wie die Entlastung des Vorstandes. Mit dem Finanzamt hatte ich diesbezüglich bereits telefonisch Kontakt aufgenommen. Eine Verlängerung kann erst beantragt werden, wenn die Frist verstrichen ist. Man hat mir im Telefonat angedeutet, dass man auf unsere derzeitige schwierige Situation Rücksicht nehmen wird.

Betriebsprüfung nach § 28p SGB IV

Im Winter hatte ich damit zu tun, die Unterlagen für die Betriebsprüfung für die Jahre 2020 bis 2023 zusammenzustellen. Diese Prüfung ist nun ohne Beanstandungen erfolgt (das Ergebnis lag zum Zeitpunkt der Mitgliederversammlung noch nicht vor) und beendet.

Abschließende Bemerkung

Ich würde gerne ausführlicher berichten, ich muss aber mit meiner Zeit gut haushalten. Vorerst konzentriere ich mich primär auf die Einführung von VEWA, den Bankeinzug der Mitgliedsbeiträge, die Finanzbuchhaltung der Jahre 2023/2024, das Aufarbeiten der Kasse und natürlich den laufenden Betrieb.

Ich bedanke mich bei allen Vorstandsmitgliedern. Ich fühlte mich jederzeit gut unterstützt. Zudem bedanke ich mich auch bei allen Mitgliedern für die konstruktive Kritik und den großen Zuspruch, den ich für meine Arbeit erhielt.

Mit Wünschen, Kritik und Fragen zum Thema Vereinsfinanzen kontaktieren Sie mich gerne: doris@dante.de.

Einladung zur BayT_EX 2024 in Neu-Ulm

Leah Neukirchen

Auch dieses Jahr gibt es einen *Bayrischen T_EX-Stammtisch*. Dieser findet am



*Samstag, den 7. September 2024
im Temporärhaus Neu-Ulm
<https://temporaerhaus.de/das-haus/>*

statt. Es freut uns, zum ersten Mal seit 2009 wieder in der Region Ulm zu sein!

Aufruf zum Einreichen von Beiträgen

Die BayT_EX hat – wie für eine T_EX-Konferenz üblich – einen Vortragsblock. Hier sind wir auf die Unterstützung von Teilnehmenden angewiesen und bitten um Einreichungen von Vorträgen per eMail an leah@vuxu.org.

Üblicherweise ist die BayT_EX an den Bedürfnissen von Anwender*innen orientiert, grundsätzlich sind Einreichungen aber aus allen Bereichen der T_EXnischen Welt willkommen (T_EX, L^AT_EX, ConT_EXt, Schriftsatz, Typografie und Druckereiwesen). Die Vorträge haben üblicherweise eine Länge von 30 Minuten, wir können allerdings auch einige längere Vorträge mit 60 Minuten einbauen.

Anmeldung



Der Eintritt ist frei. Zur besseren Planung wird allerdings um eine Anmeldung unter <https://tickets.fs.lmu.de/GAF/baytex2024/> gebeten.

Anfahrt



Das *Temporärhaus* befindet sich im ehemaligen Sport-Sohn-Gebäude, in der Augsburger Straße 23–25 in 89231 Neu-Ulm, <https://www.openstreetmap.org/?mlat=48.39632&mlon=10.00204#map=19/48.39632/10.00204>. Es ist gut zu Fuß vom Bahnhof Neu-Ulm zu erreichen (10 min) oder auch von Ulm aus mit dem Stadtbus.

Der Zugang zum Vortragsraum ist ebenerdig und barrierefrei. Ein barrierefreies WC ist vorhanden.

Ablaufplan

Das Programm steht noch nicht genau fest, aber die Veranstaltung wird ungefähr von 10 bis 18 Uhr stattfinden. Anschließend gibt es ein gemeinsames Abendessen in einem nahegelegenen Lokal. Bitte bei der Anmeldung angeben, ob ein Platz mitreserviert werden soll.

Zur Geschichte: Der Bayrische T_EX-Stammtisch

Seit 2003 veranstalten die T_EX-Stammtische aus München und Erlangen (2006 bis 2009 in Zusammenarbeit mit dem T_EX-Stammtisch Ulm) jährlich den Bayrischen T_EX-Stammtisch (aka »BayT_EX«), ein Treffen bestehend aus einem Nachmittag mit Fachvorträgen und einem anschließendem Grillabend.

Mehr über BayT_EX könnt ihr auf der Archivseite <https://www.ks-ingenieurconsult.de/TeX/Stammtisch.html> erfahren.



Kontakt

BayT_EX 2024 wird von Leah Neukirchen organisiert. Bei Fragen gerne eine Mail an leah@vuxu.org schicken.

Bretter, die die Welt bedeuten

LaTeX und Schulphysik 7: Optik

Keno Wehr

Der siebte Teil der Artikelreihe zur Schulphysik beschäftigt sich mit grafischen Darstellungen zur Strahlen- und Wellenoptik. Er stellt die Pakete `pst-optic`, `pgf-interference`, `pst-diffraction`, `tikz-optics` und `pst-optexp` vor.

Abbildungen an Linsen mit `pst-optic`

Strahlengänge bei Abbildungen an Sammell- und Zerstreuungslinsen können mit dem Paket `pst-optic`, das bereits vor längerer Zeit in dieser Zeitschrift vorgestellt wurde [6], dargestellt werden. Der nötige Befehl ist `\lens[Optionen]`. Eine `pspicture`-Umgebung ist nicht erforderlich, da sie von dem Befehl automatisch eingefügt wird.

Einen Überblick der wichtigsten Optionen bietet die Tabelle 1. Die Optionen `xLeft` und `xRight` legen die x -Werte des linken und rechten Bildrands fest. Während die Standardwerte -7.5 und 7.5 (Einheiten in Zentimetern) für den Abdruck auf A4-Papier passend sind, wählen wir hier geeignete Werte für das Format dieser Zeitschrift (Abb. 1).

```
\lens[xLeft=-4,xRight=7]
```

Wie weithin üblich werden ein Parallelstrahl, ein Mittelpunktstrahl und ein Brennpunktstrahl dargestellt. Zur Veränderung von Brennweite, Gegenstandsweite und Gegenstandsgröße gibt es die Optionen `focus`, `OA` und `AB`. `OA` ist negativ, wenn der Gegenstand links der Linse steht. Die Farbe der dargestellten Lichtstrahlen kann mit der Option `rayColor` festgelegt werden, wobei die Syntax des Pakets `xcolor` gilt (Abb. 2).

```
\lens[xLeft=-6,xRight=5,focus=1,OA=-4,AB=2,rayColor=green!50!black]
```

Option	Bedeutung	Standardwert
xLeft	x -Koordinate des linken Bildrands	-7.5
xRight	x -Koordinate des rechten Bildrands	7.5
yBottom	y -Koordinate des unteren Bildrands	-3
yTop	y -Koordinate des oberen Bildrands	3
lensType	Linsentyp (CVG bikonvexe, PCVG plankonvexe Sammellinse; DVG bikonkave, PDVG plankonkave Zerstreuungslinse)	CVG
lensHeight	Linsenhöhe	5
lensWidth	Linsendicke	0.5
focus	Brennweite (für Zerstreuungslinsen negativ zu wählen)	2
OA	Gegenstandsweite (für Gegenstände links der Linse negativ zu wählen)	-3
AB	Gegenstandsgröße	1
rayColor	Farbe der Lichtstrahlen	red
drawing	Zeichnen der Lichtstrahlen	true
nameA	Bezeichnung des Fußpunkts des Gegenstands	A
nameB	Bezeichnung der Spitze des Gegenstands	B
nameAi	Bezeichnung des Fußpunkts des Bildes	A'
nameBi	Bezeichnung der Spitze des Bildes	B'
nameF	Bezeichnung des linken Brennpunkts	F
nameFi	Bezeichnung des rechten Brennpunkts	F'
nameO	Bezeichnung des Linsenmittelpunkts	O
spotA	Position der Bezeichnungen in Grad	270
spotB		90
spotAi		90
spotBi		270
spotF		270
spotFi		270
spotO		225

Tab. 1: Paket `pst-optic`: ausgewählte Optionen des Befehls `\lens`

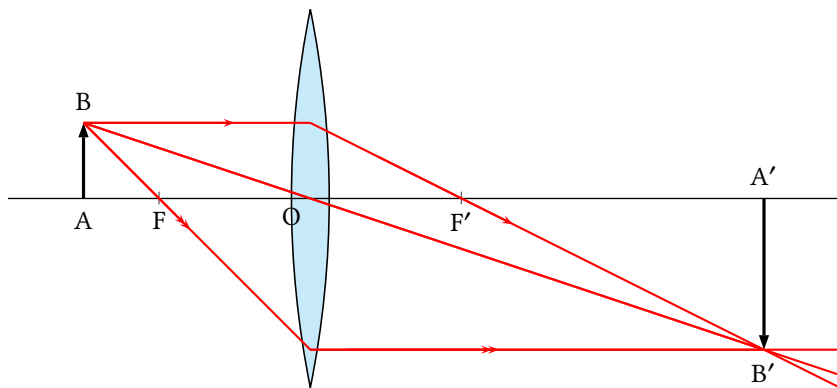


Abb. 1: Paket pst-optic: Strahlengang an einer Sammellinse

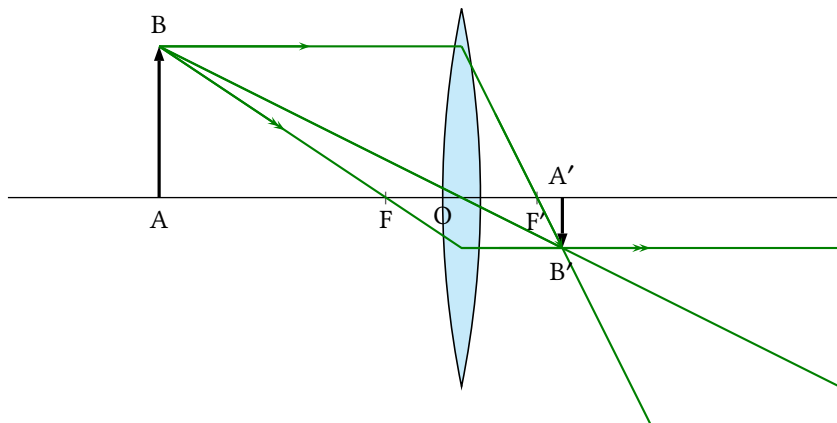


Abb. 2: Paket pst-optic: Strahlengang an einer Sammellinse mit veränderten Parametern

Für eine Zerstreuungslinse wählt man `lensType=DVG` und zusätzlich einen negativen Wert für `focus`. Die Lage der Brennpunkte wird für Zerstreuungslinsen jedoch nicht angezeigt (Abb. 3).

```
\lens[xLeft=-6,xRight=5,lensType=DVG,focus=-3,OA=-4,AB=2]
```

Um die Darstellung der Bezeichnungen zu verbessern, nutzen wir die Optionen `spotAi`, `spotBi` und `nameO` (Abb. 4).

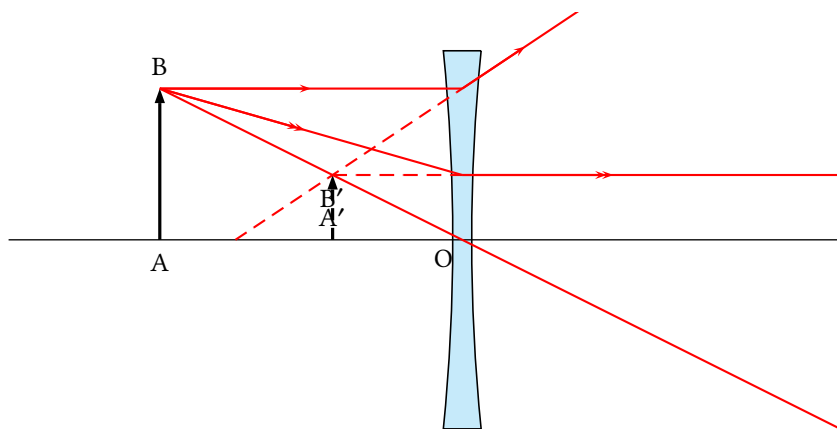


Abb. 3: Paket pst-optic: Strahlengang an einer Zerstreuungslinse

```
\lens[xLeft=-6,xRight=5,lensType=DVG,focus=-3,OA=-4,AB=2,spotAi=270,spotBi=90,
  \carrowname0={}]
```

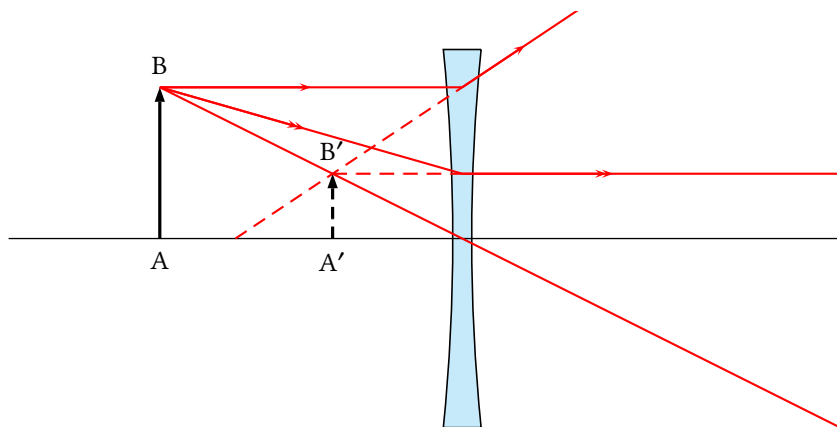


Abb. 4: Paket pst-optic: Strahlengang an einer Zerstreuungslinse mit verbesserter Beschriftung

Wenn die Abbildung um individuelle Elemente ergänzt werden soll, muss eine pspicture-Umgebung verwendet und der `\lens`-Befehl in ein `\rput` eingeschlossen werden (Abb. 5).

```
\usepackage{pst-optic}
```

```
\usepackage{siunitx}
```

```
\begin{pspicture}(-5.5,-3)(5.5,3)
```

```
\rput(0,0){\lens[xLeft=-5.5,xRight=5.5,focus=1,OA=-3,AB=2,nameA={},nameB={},  
  \cnameAi={},nameBi={},spot0=270]}
```

```
\rput[r](-3.2,1){$G=\qty{10}{\cm}$}
```

```
\rput[1](1.7,-0.5){$B=\qty{5}{\cm}$}
```

```
\end{pspicture}
```

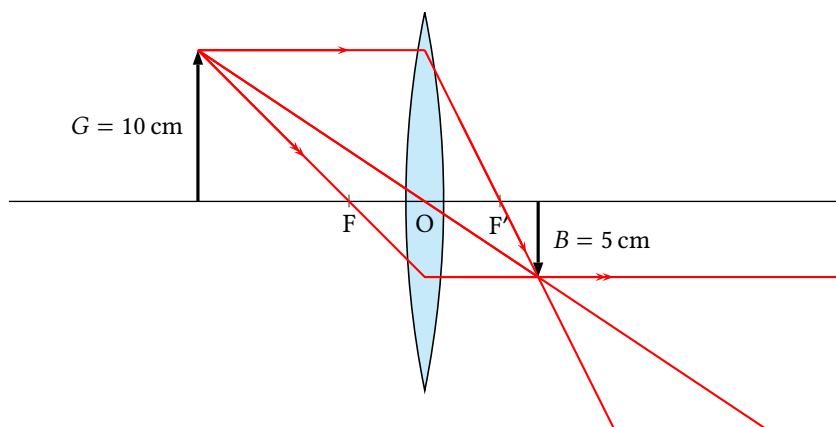


Abb. 5: Paket pst-optic: Strahlengang an einer Sammellinse mit individuellen Beschriftungselementen

Wenn bei einer Sammellinse die Gegenstandsweite kleiner als die Brennweite ist, entsteht ein virtuelles Bild auf der Gegenstandsseite (Lupeneffekt). Die Darstellung des Strahlengangs in Abb. 6 ist jedoch nicht überzeugend.

```
\lens[xLeft=-6,xRight=5,focus=4,OA=-2.2,AB=0.7,spotAi=270,spotBi=90,spot0=270]
```

Mit der Option `drawing=false` kann der ansonsten immer angezeigte Strahlengang ausgeblendet werden. Der Befehl `\Parallel(A)(B)(C){D}` zeichnet eine Parallele zur Strecke $\overline{AB}$, die im Punkt C beginnt, wobei der Endpunkt D der parallelen Strecke berechnet wird. `\psOutLine[length=l](A)(B){C}` zeichnet eine Linie der Länge l , die die Richtung AB hat und bei B beginnt, wobei der Endpunkt C berechnet wird. Wir nutzen diese Befehle, um ein Bündel paralleler Lichtstrahlen zu zeichnen, die sich in der Brennebene schneiden. Dabei kommt der Befehl `\multido` aus dem gleichnamigen Paket zum Einsatz, das von pst-optic bereits geladen wird (Abb. 7).

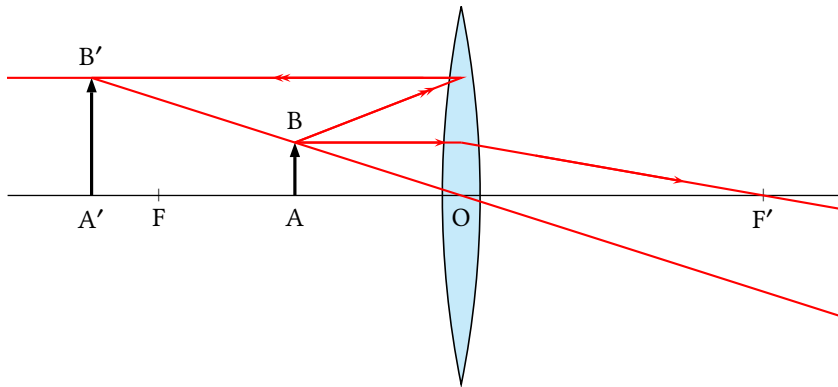


Abb. 6: Paket pst-optic: fehlerhafte Darstellung des Strahlengangs an einer Lupe

```
\begin{pspicture*}(-5,-3.5)(5,3.5)
\nodes(2,-2){FF}(-5,5){A}(0,0){O}
\multido{\ry=-2+0.5}{9}{%
  \Parallel[linecolor=red,length=8](O)(A)(0,\ry){P1}
  \psline[linecolor=red](0,\ry)(FF)
  \psOutLine[linecolor=red,length=5](0,\ry)(FF){P2}}
\rput(0,0){\lens[drawing=false]}
\psset{linewidth=0.5pt}
\psline(-5,0)(5,0)
\psline[linestyle=dashed](2,3.5)(2,-3.5)
\uput[-45](2,0){$F$}
\end{pspicture*}
```

Während die hier vorgestellten Beispiele von idealen Linsen ausgehen, können mit dem Befehl `\lensSPH` die Abbildungsfehler realer sphärischer Linsen berücksichtigt werden. Zudem können mit `pst-optic` die Strahlengänge bei Abbildungen an Hohlspiegeln dargestellt werden. Hierfür sei auf die Paketanleitung verwiesen.

Ein weiteres Paket, mit dem man Abbildungen an Linsen und Hohlspiegeln darstellen kann, ist `tikz-mirror-lens`. Die Darstellungsweise dieses Pakets entspricht jedoch nicht im wünschenswerten Maße den schulischen Anforderungen, sodass `pst-optic` vorzuziehen ist.

Das ebenfalls auf TikZ basierende Paket `simpleoptics` zeichnet Linsen und Hohlspiegel, jedoch ohne Darstellung des Strahlengangs.

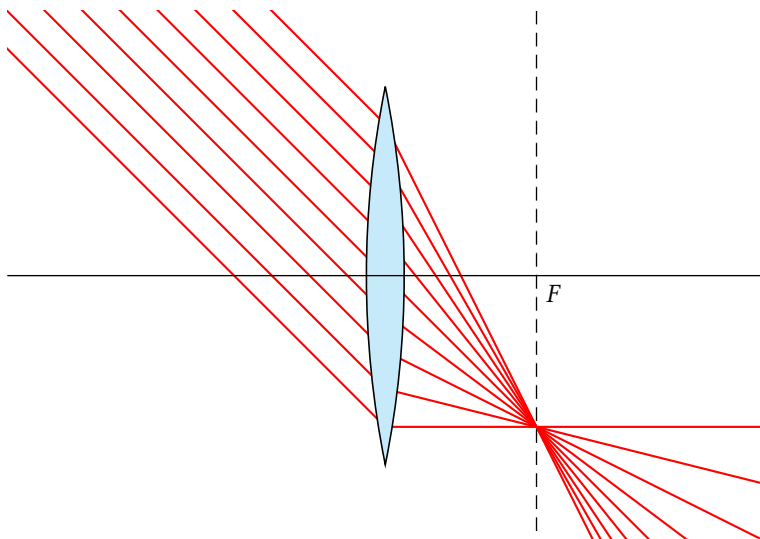


Abb. 7: Paket pst-optic: Darstellung eines Bündels paralleler Lichtstrahlen, die sich in der Brennebene schneiden

Abbildungen an Linsen mit METAPOST

Der pst-optic-Befehl `\lens` fügt Beschriftungen ein, die für schulische Darstellungen größtenteils nicht gebraucht werden, und lässt andere, die sinnvoll wären, weg. Man kann zwar die voreingestellten Beschriftungen ausblenden und manuell andere einfügen (siehe Abb. 5), doch wünscht man sich einen Befehl, der mehr den schulischen Bedürfnissen entspricht und weniger Handarbeit erfordert.

Mit den Mitteln von METAPOST lässt sich ein solcher schaffen. Wir müssen dafür zwar zunächst etwas programmieren, haben dann aber volle Flexibilität und Kontrolle. Wir beschränken uns hier auf Abbildungen an Sammellinsen, berücksichtigen aber auch virtuelle Bilder.

Mit Hilfe der Variablen `outputtemplate` legen wir `Linse1.mps`, `Linse2.mps`, ... als Namen der erzeugten Bilddateien fest. Anschließend definieren wir drei verschiedene Stiftstärken für die Lichtstrahlen, für Gegenstand und Bild sowie für Gegenstands- und Bildweite, sodann die Farbe der Lichtstrahlen.

```

1 outputtemplate := "Linse%c.mps";
2 pen strahl, bild, weite;
3 strahl = pencircle scaled 0.7bp;
4 bild = pencircle scaled 1bp;

```

```

5 weite = pencircle scaled 0.8bp;
6 color strahlfarbe;
7 strahlfarbe = red;

```

Nun folgen einige Konstanten und ein Pfad, der die Linse beschreibt und aus zwei Kurven besteht, die mit dem Operator & verbunden werden. Um das Innere des Pfades später ausfüllen zu können, ist es nötig, den Pfad mit & cycle am Ende formal zu schließen.

```

8 hL = 5cm; % Höhe der Linse
9 dL = 5mm; % Dicke der Linse
10 lM = 5bp; % Länge der Markierungsstriche
11 path linse;
12 linse = (0,-hL/2)..(-dL/2,0)..(0,hL/2) & (0,hL/2)..(dL/2,0)..(0,-hL/2) & cycle;

```

Wir definieren nun ein Makro mit fünf Argumenten: dem minimalen und dem maximalen x -Wert, der Brennweite, der Gegenstandsweite und der Gegenstandsgröße. Wir füllen die Linse in einem bläulichen Farbton, umranden sie schwarz und zeichnen die optische Achse.

```

13 def abbildung_sammellinse(expr xmin, xmax, f, g, G) =
14   fill linse withcolor (0.2*blue+0.1*green+0.7*white);
15   draw linse;
16   draw (xmin,0)--(xmax,0);
17   draw (-f,-lM/2)--(-f,lM/2);
18   draw (0,-lM/2)--(0,lM/2);
19   draw (f,-lM/2)--(f,lM/2);

```

Nun folgt die Berechnung der für die Lichtstrahlen benötigten Punkte. Hier liegt eine typische Anwendung der whatever-Funktionalität von METAPOST vor, mit der zwei Vektoren als kollinear festgelegt werden können.

```

20 z1=(-g,0); % Fuß des Gegenstands
21 z2=(-g,G); % Spitze des Gegenstands
22 z3=whatever*z2; % Ende des Mittelpunktstrahls
23 x3=xmax;
24 z4=(0,y2); % Schnittpunkt des Parallelstrahls mit der Linse
25 z5-z4=whatever*(z4-(f,0)); % z5: Ende des Brennstrahls rechts der Linse
26 x5=xmax;
27 z6-z2=whatever*(z2-(-f,0)); % z6: Ende des Brennstrahls links der Linse
28 x6=0;
29 z7=(xmax,y6); % Ende des Parallelstrahls rechts der Linse
30 z8=whatever*z2; % Spitze des Bildes
31 y8=y6;
32 z9=(x8,0); % Fuß des Bildes

```

Die Lichtstrahlen können jetzt gezeichnet werden. Falls die Gegenstandsweite kleiner als die Brennweite ist, entsteht ein virtuelles Bild. Der scheinbare Lichtweg wird gestrichelt dargestellt.

```

33 draw z2--z3 withpen strahl withcolor strahlfarbe;
34 draw z2--z4 withpen strahl withcolor strahlfarbe;
35 draw z4--z5 withpen strahl withcolor strahlfarbe;
36 draw z2--z6 withpen strahl withcolor strahlfarbe;
37 draw z6--z7 withpen strahl withcolor strahlfarbe;
38 if g < f:
39   draw z2--z8 withpen strahl dashed evenly withcolor strahlfarbe;
40   draw z4--z8 withpen strahl dashed evenly withcolor strahlfarbe;
41   draw z6--z8 withpen strahl dashed evenly withcolor strahlfarbe;
42 fi

```

Schließlich folgt die Darstellung des Gegenstands und des Bildes durch einen Pfeil, die farbliche Markierung von Gegenstands- und Bildweite und die Ausgabe der Beschriftungen.

```

43 drawarrow z1--z2 withpen bild;
44 drawarrow z9--z8 withpen bild;
45 draw origin--z9 withpen weite withcolor 0.5*green; % Bildweite
46 draw z1--origin withpen weite withcolor 0.5*blue; % Gegenstandsweite
47 label.bot(btex $F_1$ etex,(-f,-lM/2));
48 label.bot(btex $F_2$ etex,(f,-lM/2));
49 label.top(btex $g$ etex,0.5*z1);
50 label.if g<f: bot else: top fi(btex $b$ etex,0.5*z9);
51 label.lft(btex $G$ etex,0.5[z1,z2]);
52 label.if g<f: lft else: rt fi(btex $B$ etex,0.5[z8,z9]);
53 enddef;

```

Wir erzeugen nun noch je eine konkrete Darstellung mit reellem (Abb. 8) und virtuellem Bild (Abb. 9).

```

54 beginfig(1);
55   abbildung_sammellinse(-5cm,6cm,2cm,3.5cm,1.5cm);
56 endfig;
57 beginfig(2);
58   abbildung_sammellinse(-4.5cm,3.5cm,2cm,1.1cm,1cm);
59 endfig;
60 end

```

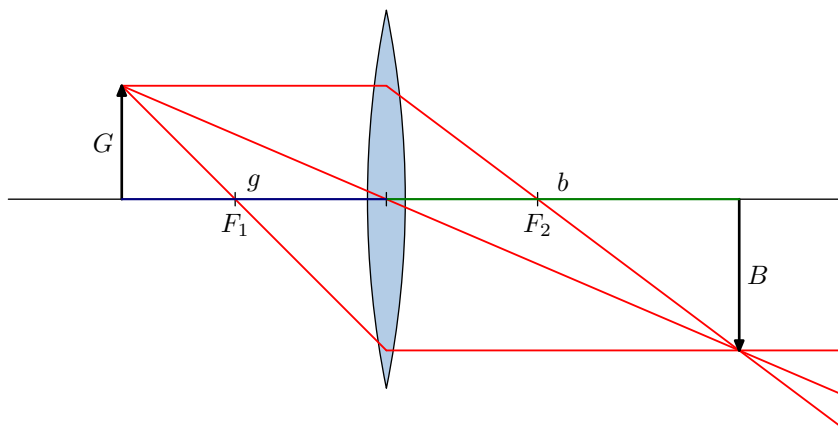


Abb. 8: Entstehung eines reellen Bilds an einer Sammellinse, gezeichnet mit METAPOST

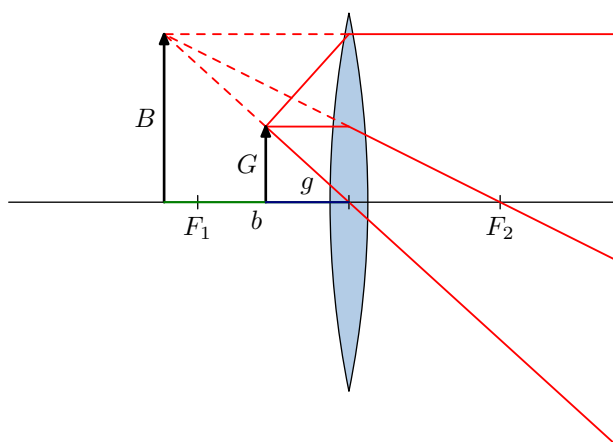


Abb. 9: Entstehung eines virtuellen Bilds an einer Sammellinse, gezeichnet mit METAPOST

Lichtbrechung mit pst-optic

Grundlegende Unterstützung bei der Darstellung der Lichtbrechung bietet pst-optic in Form des Befehls `\refractionRay(A_1)(A_2)(B_1)(B_2){n_1}{n_2}{C}`. Dabei sind A_1 und A_2 zwei definierte Punkte, die die Richtung des Lichtstrahls im ersten Medium festlegen, B_1 und B_2 zwei definierte Punkte, deren Verbindungslinie die Mediengrenze festlegt, n_1 und n_2 die Brechzahlen der beiden Medien und C der Name des Punktes, in dem der Lichtstrahl auf die Mediengrenze trifft. Dieser Punkt wird durch den Befehl berechnet, ebenso ein weiterer Punkt C' , der zusammen mit C die Richtung des gebrochenen bzw. total reflektierten Lichtstrahls festlegt. Der Befehl berechnet lediglich die Punkte C und C' ; das Zeichnen des Lichtwegs muss separat mit geeigneten Befehlen wie `\psline` erfolgen.

Das folgende Beispiel demonstriert das Grundprinzip anhand von zwei Lichtstrahlen, die aus einem Medium der Brechzahl 1,5 auf ein Medium der Brechzahl 1 treffen. Der zweite Strahl, dessen Richtung durch die Punkte H und I festgelegt ist, trifft so flach auf die Mediengrenze, dass er nicht gebrochen, sondern total reflektiert wird (Abb. 10).

```
\begin{pspicture}[showgrid](0,-1)(8,2)
\pnodes(0,0){A}(8,0){B}
\pspolygon[fillstyle=solid,fillcolor=lightgray,linecolor=lightgray](A)(B)(8,2)
\c(0,2)(A)
\psline(A)(B)
\pnodes(1,2){E}(2,0){F} % 1. Lichtstrahl
\refractionRay(E)(F)(A)(B){1.5}{1}{G}
\psset{linewidth=1pt}
\psline[linecolor=red](E)(G)
\psline[linecolor=red](G)(G')
\pnodes(3,2){H}(6,0){I} % 2. Lichtstrahl
\refractionRay(H)(I)(A)(B){1.5}{1}{J}
\psline[linecolor=blue](H)(J)
\psline[linecolor=blue](J)(J')
\end{pspicture}
```

Das nächste Beispiel stellt den Durchgang von zwei Lichtstrahlen durch einen rechteckigen Glaskörper dar. Wir verwenden zwei weitere Befehle, die von pst-optic bereitgestellt werden. `\arrowLine(A)(B){m}` zeichnet eine Linie von A nach B , die mit m Pfeilspitzen versehen ist; der Befehl `\psOutLine` wurde bereits auf S. 14 beschrieben (Abb. 11).

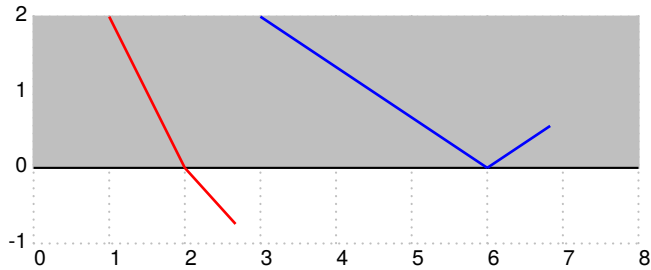


Abb. 10: Paket pst-optic: Brechung und Totalreflexion beim Übergang zum optisch dünneren Medium

```
\begin{pspicture}[showgrid](-4,-3)(4,3)
\pnodes(-2,-1){A}(2,-1){B}(2,1){C}(-2,1){D}
\uput[-135](A){A}\uput[-45](B){B}\uput[45](C){C}\uput[135](D){D}
\pspolygon[fillcolor=lightgray,fillstyle=solid](A)(B)(C)(D)
\psset{linewidth=2pt,arrowsize=5pt,arrows=->}
\pnodes(-3,3){E1}(-2,2){F1}
\refractionRay(E1)(F1)(C)(D){1}{1.5}{G1} % Brechung an der Oberkante
\psset{linecolor=red}
\arrowLine(E1)(G1){2}
\refractionRay(G1)(G1')(A)(B){1.5}{1}{H1} % Brechung an der Unterkante
\arrowLine(G1)(H1){1}
\arrowLine(H1)(H1'){1}
\psOutLine[length=1.5](H1)(H1'){I1}
\pnodes(4,2){E2}(3,1.2){F2}
\refractionRay(E2)(F2)(B)(C){1}{1.5}{G2} % Brechung an der rechten Kante
\psset{linecolor=blue}
\arrowLine(E2)(G2){2}
\refractionRay(G2)(G2')(A)(B){1.5}{1}{H2} % Totalreflexion an der Unterkante
\arrowLine(G2)(H2){2}
\refractionRay(H2)(H2')(A)(D){1.5}{1}{I2} % Brechung an der linken Kante
\arrowLine(H2)(I2){1}
\arrowLine(I2)(I2'){1}
\psOutLine[length=1.5](I2)(I2'){J2}
\end{pspicture}
```

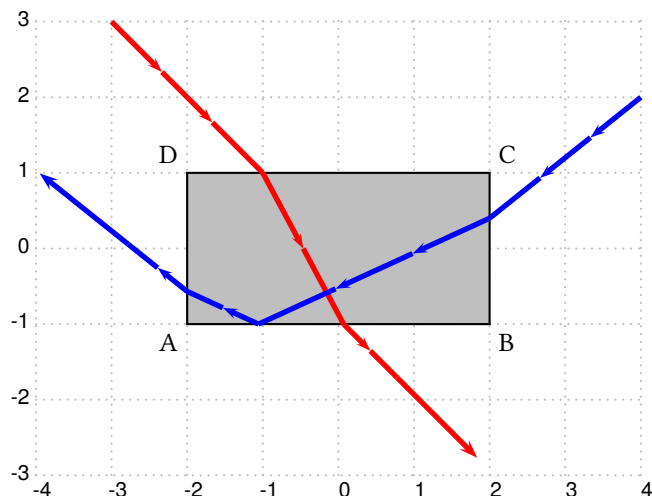


Abb. 11: Paket pst-optic: Brechung und Totalreflexion beim Durchgang durch einen rechteckigen Glaskörper

Interferenzbilder mit pgf-interference

Zu den eindrucklichsten Belegen für die Wellennatur des Lichts zählt das Auftreten von Interferenzmustern nach dem Durchgang durch feine Spaltanordnungen, die im Physikunterricht der gymnasialen Oberstufe untersucht werden.

Da in Physiksammlungen nur ein begrenzter Vorrat an Lichtquellen und Beugungsobjekten zur Verfügung steht, ist es wünschenswert, auch die Interferenzbilder für Wellenlängen und Spaltabmessungen, die nicht experimentell zugänglich sind, präsentieren zu können.

Das Paket pgf-interference macht dies möglich. Unterstützt werden nur monochromatisches Licht und regelmäßige Spaltanordnungen (Einzel-, Doppel- und Mehrfachspalte). Auch können bisher nur Interferenzstreifen dargestellt werden, wie sie bei vertikaler Aufweitung des Lichts entstehen, nicht die typischen „Interferenzkuller“, die von einem Laserstrahl erzeugt werden.

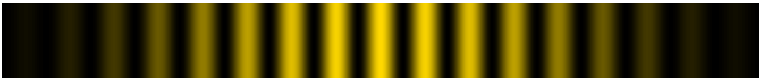
Definiert werden zwei Befehle: Mit `\pgfinterferencepattern{Optionen}` wird ein Interferenzmuster erzeugt. Mit `\pgfinterferenceoptions{Optionen}` werden Optionen festgelegt, ohne ein Interferenzmuster zu zeichnen. Das Argument ist jeweils eine kommaseparierte Optionenliste. Die verfügbaren Optionen sind in Tabelle 2 aufgeführt.

Option	Bedeutung	Standardwert
wavelength	Wellenlänge des Lichts in m. Die Voreinstellung entspricht der Wellenlänge des Helium-Neon-Lasers. Der Maximalwert ist 1.	632.8e-9
intensity	relative Lichtintensität	1.0
slits	Anzahl der Spalten	2
slit-distance	Spaltabstand in m	1e-4
slit-width	Spaltbreite in m	1e-5
screen-distance	Abstand des Schirms vom Beugungsobjekt in m	1.0
screen-width	Breite des Schirms in m	0.1
screen-height	Höhe des Schirms in m	0.03
scale	Faktor, um den der Schirm in der Abbildung skaliert wird	1.0
screen-color	Farbe des Schirms in xcolor-Syntax	black
ruler	Zeigt ein Lineal mit Zentimeterskala an. Mögliche Werte sind above, below, screen (das Lineal dient dann selbst als Schirm) und none.	none

Tab. 2: Paket pgf-interference: Optionen für Interferenzmuster

Standardmäßig wird das Interferenzmuster eines Doppelspalts simuliert. Die Farbe der Interferenzmaxima wird mit Hilfe des Pakets xcolor aus der Wellenlänge bestimmt. Alle Längen müssen in Metern ohne Maßeinheit angegeben werden, wobei die wissenschaftliche Schreibweise mit einem kleinen e verwendet werden kann.

```
\pgfinterferencepattern{wavelength=590e-9,screen-height=0.01}
```



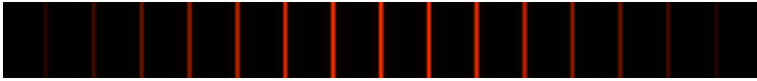
Für Wellenlängen außerhalb des sichtbaren Bereichs werden die Interferenzmaxima schwarz dargestellt. Um solche Maxima sichtbar zu machen, sollte weiße Schirmfarbe eingestellt werden.

```
\pgfinterferencepattern{wavelength=850e-9,screen-height=0.01,screen-color=white
↪}
```



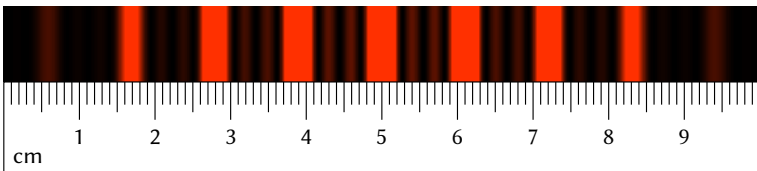
Um ein optisches Gitter zu simulieren, genügt es, eine Spaltanzahl von 10 zu wählen.

```
\pgfinterferencepattern{slits=10,screen-height=0.01}
```



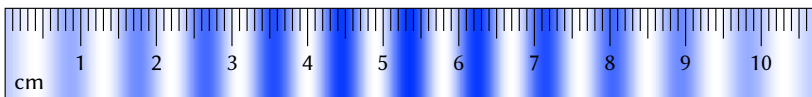
Zur quantitativen Auswertung kann oberhalb oder unterhalb des Interferenzmusters ein Lineal eingeblendet werden. Um Nebenmaxima besser sichtbar zu machen, sollte mit der Option `intensity`, die den Standardwert 1 hat, die Lichtintensität erhöht werden.

```
\pgfinterferencepattern{slits=4,slit-distance=2e-4,slit-width=4e-5,screen-
  \rightarrow distance=3.5,screen-height=0.01,intensity=5,ruler=below}
```



Das Lineal kann auch selbst als Schirm fungieren.

```
\pgfinterferencepattern{wavelength=450e-9,screen-distance=2,ruler=screen,screen
  \rightarrow width=0.107}
```



Für Lichtquellen mit kontinuierlichem Spektrum oder mehreren Spektrallinien können leider keine Interferenzmuster erzeugt werden.

Das Übersetzen von Dokumenten mit vielen Interferenzmustern kann aufgrund der nötigen Berechnungen lange dauern. Daher gibt es einen Entwurfsmodus, in dem nur der Schirm, aber kein Interferenzmuster gezeichnet wird, was wesentlich schneller geht. Der Entwurfsmodus wird durch die Paketooption `draft` aktiviert.

```
\usepackage[draft]{pgf-interference}
```

Interferenzbilder mit pst-diffraction

Eine Ergänzung zu pgf-interference bietet das Paket pst-diffraction, das Beugungs-/Interferenzbilder für zweidimensionale Blendenöffnungen zeichnet, die die Form eines Rechtecks, Kreises oder gleichseitigen Dreiecks haben. Im Falle von Rechteck und Kreis können auch Interferenzbilder für zwei gleichartige Öffnungen berechnet werden. Zudem kann die Intensitätsverteilung des Interferenzmusters durch eine 3D-Grafik veranschaulicht werden. Wie bei pgf-interference wird die Farbe des Musters aus der Wellenlänge ermittelt, es wird nur monochromatisches Licht unterstützt und die Berechnung kann lange dauern.

Die wichtigsten Optionen der von pst-diffraction bereitgestellten Befehle sind in Tabelle 3 aufgeführt. Beugungsbilder von rechteckigen Öffnungen werden mit dem Befehl `\psdiffractionRectangle[Optionen]`, der innerhalb einer pspicture-Umgebung verwendet wird, erstellt (Abb. 12).

```
\begin{pspicture}(-2.5,-2.5)(2.5,2.5)
\psdiffractionRectangle[f=2]
\end{pspicture}
\hfill
\begin{pspicture}(-2.5,-2)(2.5,3)
\psdiffractionRectangle[f=1.6,IIID,Alpha=30,Beta=50]
\end{pspicture}
```

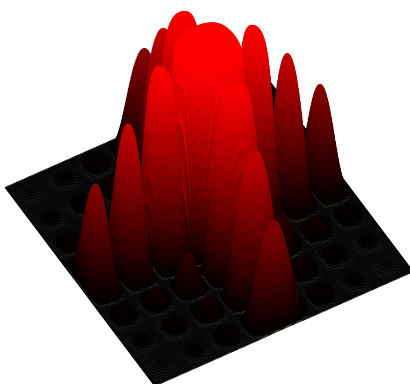
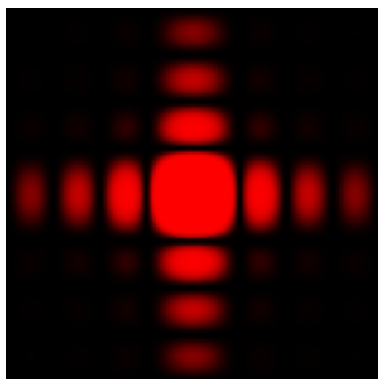


Abb. 12: Paket pst-diffraction: Beugungsmuster hinter einer quadratischen Öffnung

Die relative Lichtintensität lässt sich über die Option contrast einstellen (Abb. 13).

Option	Bedeutung	Standardwert
a	Breite der rechteckigen Öffnung in m	2e-4
k	Faktor, um den die Höhe der rechteckigen Öffnung von der Breite abweicht	1
twoSlit	zwei rechteckige Beugungsöffnungen	false
s	Abstand der rechteckigen Beugungsöffnungen in m, nur relevant für twoSlit=true	1.2e-2
r	Radius der kreisförmigen Öffnung in m	1e-3
twoHole	zwei kreisförmige Beugungsöffnungen	false
d	Abstand der kreisförmigen Beugungsöffnungen in m, nur relevant für twoHole=true	6e-3
showFunc	Anzeige eines Diagramms mit dem Intensitätsverlauf in Abhängigkeit vom Radius	false
h	Höhe der dreieckigen Beugungsöffnung	5e-4
f	Brennweite einer hinter dem Beugungsobjekt aufgestellten Linse in m; bestimmt die Größe des Beugungsmusters	5
lambda	Wellenlänge des Lichts in nm	650
pixel	Auflösung	0.5
contrast	relative Lichtintensität; die Voreinstellung entspricht dem Maximalwert.	38
IIID	3D-Darstellung	false
Alpha	Azimutwinkel der Blickrichtung bei 3D-Darstellung in Grad	45
Beta	Höhenwinkel der Blickrichtung bei 3D-Darstellung in Grad	30

Tab. 3: Paket pst-diffraction: ausgewählte Optionen für Beugungs-/Interferenzmuster

```
\begin{pspicture}(-2.5,-1.5)(2.5,1.5)
\psdiffractioRectangle[k=2,f=2,contrast=6]
\end{pspicture}
\hfill
\begin{pspicture}(-2.5,-2)(2.5,2)
\psdiffractioRectangle[k=2,f=2,contrast=6,IIID,Alpha=30,Beta=80]
\end{pspicture}

\begin{pspicture}(-2.5,-1.5)(2.5,1.5)
\psdiffractioRectangle[k=2,f=2,contrast=24]
\end{pspicture}
\hfill
\begin{pspicture}(-2.5,-2)(2.5,2)
\psdiffractioRectangle[k=2,f=2,contrast=24,IIID,Alpha=30,Beta=80]
\end{pspicture}
```

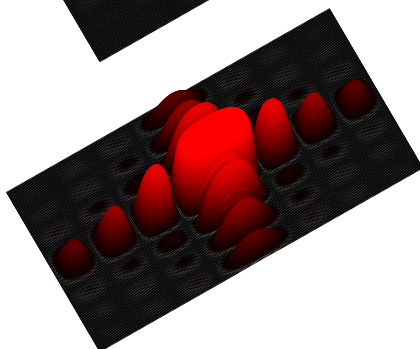
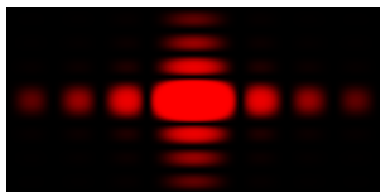
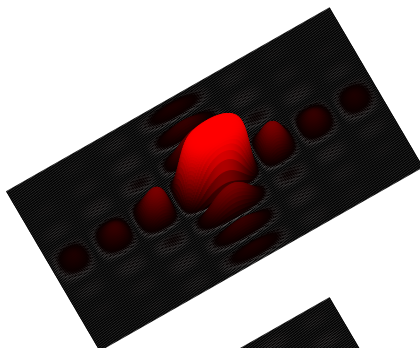
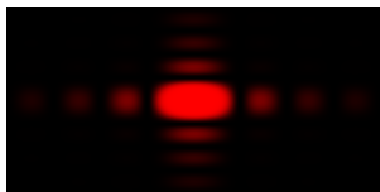


Abb. 13: Paket pst-diffraction: Beugungsmuster hinter einer rechteckigen Öffnung bei verschiedenen Lichtintensitäten

`\psdiffractiCircular[Optionen]` (Abb. 14) gibt es für Beugungsbilder kreisförmiger Öffnungen, `\psdiffractiTriangle[Optionen]` für dreieckige Öffnungen.

```
\begin{pspicture}(-3,-3)(3,3)
\psdiffractiCircular[r=0.5e-3,f=9,pixel=0.5,lambda=520]
\end{pspicture}
\hfill
\begin{pspicture}(-2.5,-1.5)(2.5,4)
\psdiffractiCircular[r=0.5e-3,f=8,pixel=0.5,lambda=520,IIID]
\end{pspicture}
```

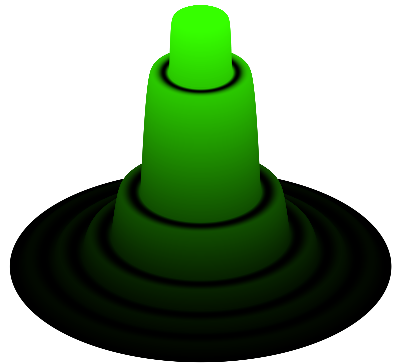
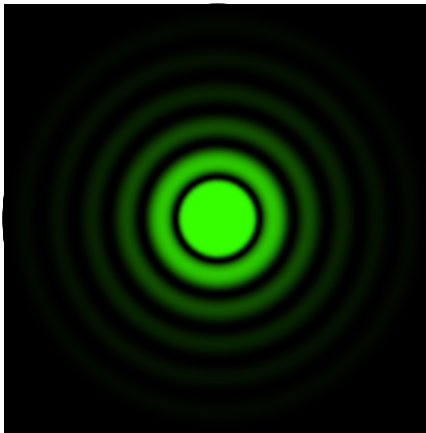


Abb. 14: Paket `pst-diffraction`: Beugungsmuster hinter einer kreisförmigen Öffnung

Die Option `twoSlit` erlaubt die Berechnung des Interferenzmusters für zwei rechteckige Öffnungen. Interessant ist hier ein Vergleich mit dem Muster von (vertikal unendlich ausgedehnten) Spalten (Abb. 15).

```
\usepackage{pgf-interference}
\usepackage{pst-diffraction}
```

```
\begin{pspicture}(-5.5,-0.5)(5.5,0.5)
\psdiffractiRectangle[twoSlit,lambda=550,a=1e-4,s=3e-4,f=2.5,k=10]
\end{pspicture}

\smallskip
```

```
\pgfinterferencepattern{wavelength=550e-9,slit-width=1e-4,slit-distance=3e-4,
  \>screen-distance=2.5,intensity=20,screen-width=0.11,screen-height=0.01}
```

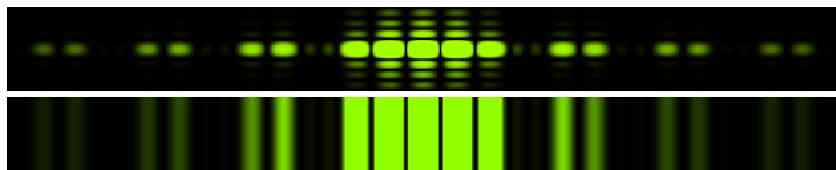


Abb. 15: Pakete pst-diffraction und pgf-interference: Vergleich der Interferenzmuster für zwei vertikale Spalte (unten) und zwei Rechtecköffnungen (oben)

Optische Aufbauten

Die Pakete tikz-optics und pst-optexp ermöglichen das Zeichnen optischer Versuchsaufbauten. Die meisten Anwendungsfälle sind eher dem universitären Kontext zuzuordnen; in der gymnasialen Oberstufe können die Pakete aber beispielsweise zur Darstellung von Interferometern nützlich sein.

Bei tikz-optics handelt es sich nicht um ein LaTeX-Paket im Sinne einer sty-Datei, sondern um eine TikZ-Bibliothek, die folgendermaßen eingebunden wird:

```
\usetikzlibrary{optics}
```

Zusätzlich muss der Umgebung tikzpicture die Option use optics übergeben werden, um die Elemente der Bibliothek nutzen zu können.

Nun folgt der vollständige Code für ein Michelson-Interferometer. Aus Gründen der didaktischen Reduktion wird auf die Darstellung von Linsen verzichtet. Zur Markierung des Lichtwegs verwenden wir noch die TikZ-Bibliothek decorations.shapes (Abb. 16).

```
\usepackage{tikz}
\usetikzlibrary{optics}
\usetikzlibrary{decorations.shapes}

\begin{tikzpicture}[use optics,font={\small}]
\node[mirror,rotate=90] (S1) at (0,2.5) {};
\node[above=2mm] at (S1.east) {Spiegel 1, fest};
\node[mirror] (S2) at (2.7,0) {};
\node[right=2mm] at (S2.east) {Spiegel 2, verschiebbar};
\coordinate[below left=3mm] (A) at (S2.south);
\coordinate[below right=3mm] (B) at (S2.south);
```

```

\draw[<->,blue,thick] (A)--(B);
\node[laser] (L) at (-3,0) {Laser};
\node[semi-transparent mirror,rotate=-45] (HS) at (0,0) {};
\node[below left,align=right,text width=3cm] at (HS.south) {halbdurchlässiger
  ↳Spiegel};
\node[screen,rotate=-90] (S) at (0,-2.7) {};
\node[below=2mm] at (S.east) {Schirm};
\draw[red] (L.east)--(HS)--(S1)--(HS)--(S);
\draw[red,decorate,decoration={triangles,pre length=2mm},segment length=7mm] (L.
  ↳east)--(HS)--(S1)--(HS)--(S);
\draw[red] (HS)--(S2)--(HS)--(S);
\draw[red,decorate,decoration={triangles,pre length=3.5mm},segment length=7mm]
  ↳(HS)--(S2)--(HS)--(S);
\end{tikzpicture}

```

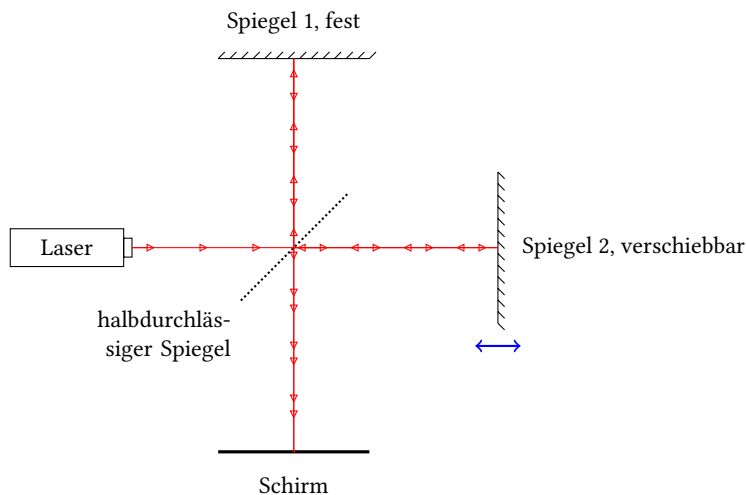


Abb. 16: Paket tikz-optics: Aufbau eines Michelson-Interferometers

Bei pst-optexp handelt es sich um ein übliches Paket, das mit `\usepackage` geladen wird. Der Code für den optischen Aufbau wird innerhalb einer `pspicture`-Umgebung in die Umgebung `optexp` eingeschlossen.

Wiederum zeichnen wir ein Michelson-Interferometer. Mit der Option `beam` können wir neben den Bauteilen auch gleich den Strahlengang zeichnen. Mit `ArrowInside=>` versehen wir die Lichtstrahlen mit Pfeilspitzen (Abb. 17).

```
\usepackage{pst-optexp}
```

```
\begin{pspicture}(-3,-3)(3,3)
\pnodes(-1.5,0){L}{(0,0){HS}{(0,2){S1}{(2,0){S2}{(0,-2){Sch}
\psset{mirrortype=extended,mirrordepth=0.2,labelstyle={\small},labeloffset=0.4,
  \c@ArrowInside=>}
\begin{optexp}
\optsource[beam,innerlabel](L)(HS){Laser}
\beamsplitter[bsstyle=plate,bssize=1,labelalign=tr,labelangle=-135,labeloffset
  \c@=>1](L)(HS)(S1){\parbox{2cm}{\raggedleft halbdurchlässiger Spiegel}}
\mirror[beam,labelalign=b](HS)(S1)(HS){Spiegel 1, fest}
\mirror[beam,variable,labelalign=l](HS)(S2)(HS){Spiegel 2, verschiebbar}
\optplate[beam,position=end,plateheight=1.2,labelalign=t,labelangle=90](HS)(Sch
  \c@){Schirm}
\end{optexp}
\end{pspicture}
```

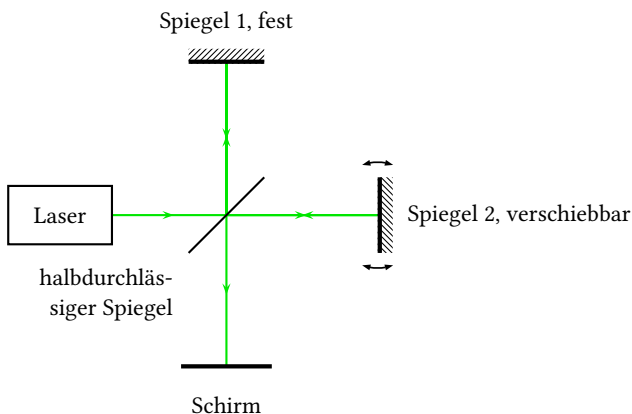


Abb. 17: Paket pst-optexp: Aufbau eines Michelson-Interferometers

Aufgrund seiner Verwandtschaft mit dem Themengebiet der Optik soll abschließend noch auf das Paket pst-moire hingewiesen werden, das umfangreiche Möglichkeiten zur Darstellung des Moiré-Effekts bietet.

Literatur

- [1] Christoph Bersch: pst-optexp, Drawing optical experimental setups, <https://mirror.clientvps.com/CTAN/graphics/pstricks/contrib/pst-optexp/pst-optexp.pdf>.

- [2] Michel Fruchart: optics — une bibliothèque tikz pour les schémas d'optique, <https://ftp.tu-chemnitz.de/pub/tex/graphics/pgf/contrib/tikz-optics/tikz-optics.pdf>.
- [3] Jürgen Gilg, Manuel Luque, Herbert Voß: pst-magneticfield -- Plotting a magnetic field with PSTricks, <https://ctan.org/tex-archive/graphics/pstricks/contrib/pst-magneticfield>.
- [4] Manuel Luque, Herbert Voß: pst-diffraction, Diffraction patterns for diffraction from circular, rectangular and triangular apertures, <https://ctan.joethei.xyz/graphics/pstricks/contrib/pst-diffraction/doc/pst-diffraction-doc.pdf>.
- [5] — pst-optic, Lenses and Mirrors, <https://mirror.physik.tu-berlin.de/pub/CTAN/graphics/pstricks/contrib/pst-optic/doc/pst-optic-doc.pdf>.
- [6] Herbert Voß: »Optische Darstellungen mit pst-optic«, 15.1 (2003), 40–59.
- [7] Keno Wehr: pgf-interference, Interferenzmuster zeichnen mit pgf/TikZ, <https://ftp.agdsn.de/pub/mirrors/latex/dante/graphics/pgf/contrib/pgf-interference/pgf-interference-de.pdf>.

Diagramme mit TikZ: Ein Beispiel

Uwe Ziegenhagen

Seit einiger Zeit beschäftige ich mich mit der Erzeugung elektronischer Musik und wollte in diesem Zusammenhang gern mit TikZ ein Blockdiagramm erstellen, das den Signalweg in einem Synthesizer zeigt. Der Weg zum fertigen Diagramm kann vielleicht auch für einige als Anregung dienen, komplexere Zeichnungen mit TikZ anzufertigen.

Grundlagen und Vorlage

Klassische Synthesizer bestehen üblicherweise aus einer Reihe von miteinander verschalteten VCOs (»Voltage-Controlled Oscillators«, spannungsgesteuerten Oszillatoren), VCFS (»Voltage-Controlled Filters«, spannungsgesteuerten Filtern) und VCAs (»Voltage-Controlled Amplifiers«, spannungsgesteuerten Verstärkern), die durch sogenannte ADSR-Hüllkurven gesteuert werden. Visualisiert man diese in einem Blockdiagramm, so ergibt sich ein Bild wie in Abbildung 1, das von der Webseite von Kent Lundberg (MIT) genommen wurde. Diese Abbildung soll die Vorlage für unsere heutige TikZ-nische Reise sein.

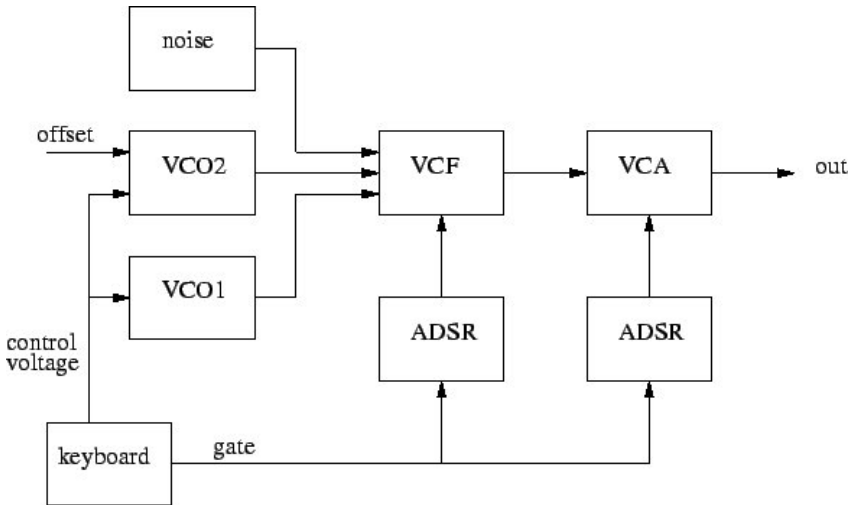


Abb. 1: Vorlage, Quelle: <http://web.mit.edu/klund/www/weblatex/node2.html>

Umsetzung in TikZ

Um eine höher aufgelöste Version der Grafik zu bekommen und damit auch die Möglichkeit zu erhalten, später einzelne Teile grafisch betonen zu können, habe ich mich für den »Nachbau« mittels TikZ entschieden. Die Umsetzung soll dabei so nah am Original sein, wie es aus meiner Sicht sinnvoll ist.

Nach dem Laden des tikz-Pakets ist mein erster Schritt beim »TikZen« stets, ein passendes Gitter zu zeichnen, das als Basis für die darauf folgenden Positionierungsarbeiten dient. TikZ kennt dazu die grid-Option des `\draw`-Befehls; Farbe, Schrittweite und Liniendicke des Gitters lassen sich leicht anpassen.

Außerdem setze ich gleich einen »Node« (»Knoten«) in die Zeichnung. Wer mit dem Begriff nichts anfangen kann, der findet in der Dokumentation unter <https://tikz.dev/tikz-shapes> die passende Erläuterung, die ich hier gern frei übersetzt wiedergeben möchte: Nodes sind in ihrer einfachsten Form – gegebenenfalls sogar leere – Zeichenketten, die an Koordinaten gesetzt werden. Die Koordinaten kann man dabei fest vorgeben – so wie ich es in diesem Beispiel mache. Man kann sie aber auch über die Nähe zu anderen Nodes berechnen. Man kann den Nodes auch Namen geben, was spätere Zeichenoperationen deutlich vereinfacht. Sie können Vorder- und Hintergrundfarben sowie eine Form haben, kurz gesagt, es sind wunderbare kleine Konstrukte, die uns das Zeichnen erleichtern.

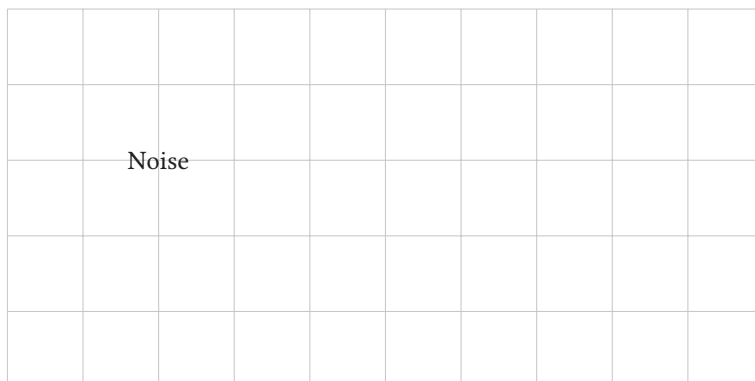


Abb. 2: Ein einzelner Node im Zeichengitter

```
\begin{tikzpicture}
\draw[step=1cm,lightgray,thin] (0,0) grid (10,5);
\node at (2,3) (noise) {Noise};
\end{tikzpicture}
```

Listing 1: Quellcode für Abbildung 2

Im nächsten Schritt kümmern wir uns um die äußere Form der einzelnen Nodes. Dazu definieren wir in den Optionen des `tikzpicture` eine Stilvorlage, die wir dann einzelnen Nodes einfach zuweisen können. Ich habe mich für eine schwarze, rechteckige Form entschieden. Die minimale Breite jedes Node wird auf 20 Millimeter, die minimale Höhe auf 10 Millimeter gesetzt, siehe Listing 2.

```
\begin{tikzpicture}[
box/.style={rectangle,thick,draw=black,minimum width=20mm, minimum height=10mm}]
\draw[step=1cm,lightgray,thin] (0,0) grid (10,5);
\node at (2,3) [box] (noise) {Noise};
\end{tikzpicture}
```

Listing 2: Quellcode für Abbildung 3

Damit haben wir alles, um die restlichen Nodes erzeugen und positionieren zu können. Es macht dabei nichts, wenn einzelne Nodes über das Grid hinausragen, da in der finalen Version der Zeichnung das Gitter sowieso nicht enthalten sein wird. Für den zweizeiligen Text im »control voltage« Node habe ich mich eines Kniffs aus <https://tex.stackexchange.com/questions/31096/how-can-i-use-linebreak-inside-a-node-in-tikz/31100#31100> bedient, der eine Parbox nutzt,

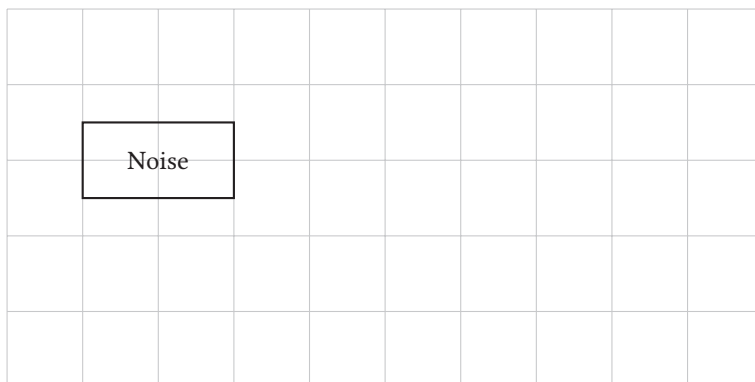


Abb. 3: Gitter und Node, mit Stilvorlage

deren Breite mittels `\widthof`-Befehl aus dem `calc`-Paket genau auf die Länge des »control«-Wortes gesetzt wird.

```
\begin{tikzpicture}[
  box/.style={rectangle,thick,draw=black,minimum width=20mm,
  minimum height=10mm,align=center},
]
\draw[step=1cm,lightgray,thin] (-1,0) grid (11,7);
\node at (2,6) [box] (noise) {Noise};
\node at (2,4.5) [box] (vco2) {VCO2};
\node at (2,3) [box] (vco1) {VCO1};
\node at (5,4.5) [box] (vcf) {VCF};
\node at (8,4.5) [box] (vca) {VCA};
\node at (5,2.5) [box] (adsr1) {ADSR};
\node at (8,2.5) [box] (adsr2) {ADSR};
\node at (0,1) [box] (keyboard) {Keyboard};

\node at (0,5) (offset) {offset};
\node at (10.5,4.5) (output) {output};
\node at (2,1.25) (gate) {gate};

\node at (-0.75,3) (cv) {\parbox{\widthof{control}}{
  control \\\ voltage}};
\end{tikzpicture}
```

Listing 3: Quellcode für Abbildung 4

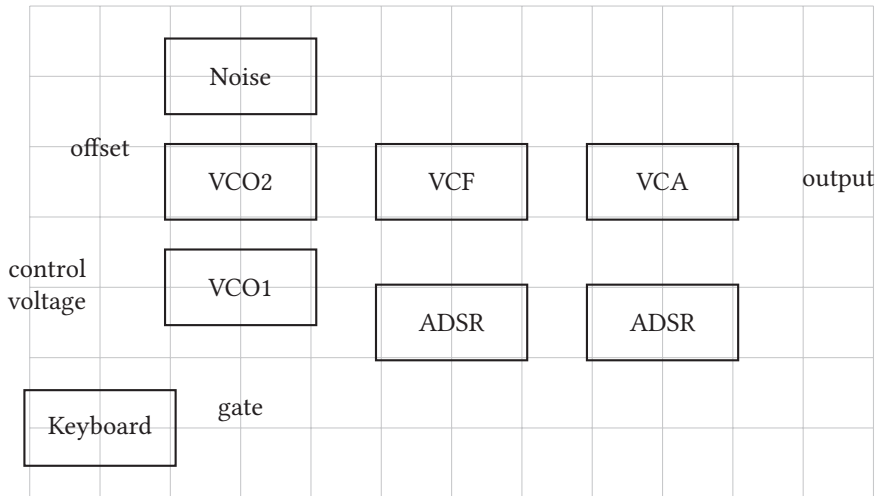


Abb. 4: Grafik mit den finalen Nodes

Bevor wir jetzt im nächsten Schritt die einzelnen Nodes untereinander mit Pfeilen verbinden, schauen wir uns die Möglichkeiten an, die TikZ uns dazu bietet. Listing 4 zeigt dazu einige der Möglichkeiten. Neben den direkten Linien (mit `--`) gibt es auch eckige Verbindungen (mit `|-` für »erst vertikal, dann horizontal« bzw. `-|` für »erst horizontal, dann vertikal«). Wir können auch die Winkel angeben, mit denen die Pfeile aus einem beziehungsweise in einen Node kommen (`g` und `h`), auch Bezier-Kurven lassen sich nutzen (`i` und `j`, die beiden zugehörigen Kontrollpunkte sind mit einem `*` markiert). Das letzte Beispiel (Nodes `k` und `l`) nutzt dann noch einige der durch TikZ vorgegebenen Anker, die über die Himmelsrichtungen adressiert werden. Siehe dazu auch die Grafik in der Anleitung unter <https://tikz.dev/library-shapes>.

```
\begin{tikzpicture}[
box/.style={rectangle,thick,draw=black,
minimum width=10mm, minimum height=10mm,align=center}]
\draw[step=1cm,lightgray,thin] (0,0) grid (8,7);

\node at (1,1) [box] (a) {a};
\node at (3,2) [box] (b) {b};
\node at (1,3) [box] (c) {c};
\node at (3,4) [box] (d) {d};
\node at (1,5) [box] (e) {e};
\node at (3,6) [box] (f) {f};
```

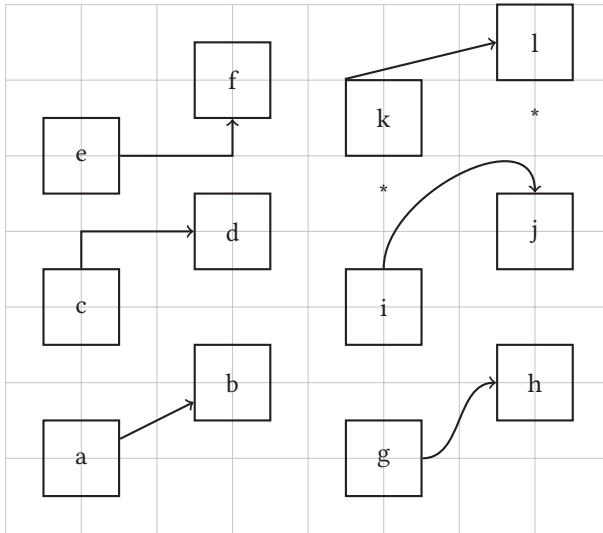


Abb. 5: Exkurs zum Verbinden von Nodes

```

\node at (5,1) [box] (g) {g};
\node at (7,2) [box] (h) {h};
\node at (5,3) [box] (i) {i};
\node at (7,4) [box] (j) {j};
\node at (5,4.5) (cp1) {*};
\node at (7,5.5) (cp2) {*};
\node at (5,5.5) [box] (k) {k};
\node at (7,6.5) [box] (l) {l};

\draw [thick,->] (a) -- (b);
\draw [thick,->] (c) -- (d);
\draw [thick,->] (e) -- (f);
\draw [thick,->] (g) to [out=0,in=180] (h);
\draw [thick,->] (i) .. controls (cp1) and (cp2) .. (j);
\draw [thick,->] (k.north west) -- (l.west);
\end{tikzpicture}

```

Listing 4: Quellcode für Abbildung 5

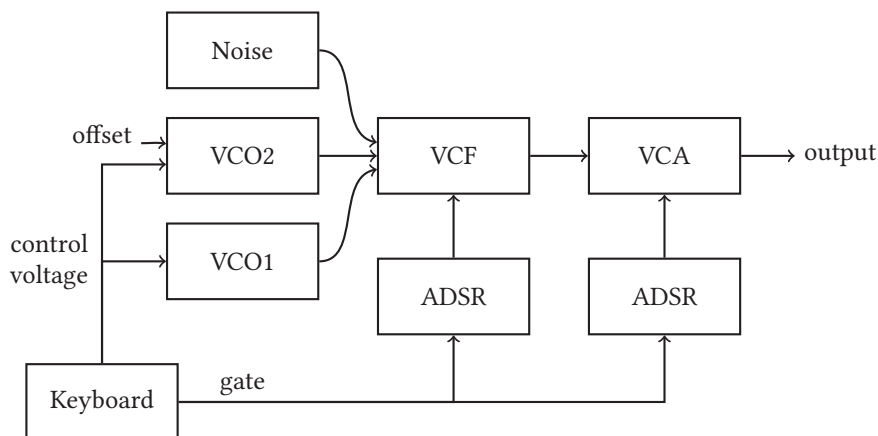


Abb. 6: Finales Diagramm

Um die einzelnen Nodes miteinander zu verbinden, setzen wir einfach weitere `\draw`-Befehle, die die Namen unserer Nodes als Parameter erhalten.

```

\draw [thick,->] (vco1) to [out=0,in=190] (vcf);
\draw [thick,->] (vco2) -- (vcf);
\draw [thick,->] (adsr1) -- (vcf);
\draw [thick,->] (adsr2) -- (vca);
\draw [thick,->] (keyboard) |- (adsr1);
\draw [thick,->] (keyboard) |- (adsr2);
\draw [thick,->] (keyboard) |- (vco1);
\draw [thick,->] (keyboard) |- (vco2);
\draw [thick,->] (noise) to [out=0,in=170] (vcf);
\draw [thick,->] (vcf) -- (vca);
\draw [thick,->] (vca) -- (output);
\draw [thick,->] (offset) to [out=0,in=170] (vco2);
  
```

Listing 5: Quellcode für Abbildung 6 (Auszug)

Fazit

TikZ verblüfft jedes Mal wieder aufs Neue, was die Erstellung von Grafiken angeht. Die Umsetzung in einem Zeichenprogramm hätte sicherlich nur einen Bruchteil der Zeit erfordert, es hätte aber mit Sicherheit nicht so viel Spaß und Aha-Effekte dabei gegeben.

Moloch: Die Wiederauferstehung des $\text{\LaTeX}$ -Beamer-Stils Metropolis ¹

Johan Larsson

Metropolis ist ein beliebter und moderner Stil für $\text{\LaTeX}$ Beamer, welcher aber unglücklicherweise nicht mehr aktiv gepflegt wird. Das hat zur Folge, dass die Liste an Problemen immer länger wird. Moloch ist ein Fork von Metropolis, der zum Ziel hat, diese Probleme zu beheben und einige neue Funktionalitäten hinzuzufügen.

So lange ich mich erinnern kann, befindet sich meine Konfiguration für Beamer-Präsentationen mit $\text{\LaTeX}$ in einem stetigen Wandel. Auf der Suche nach einem minimalistischen, aber dennoch funktionalen und optisch ansprechenden Design, habe ich viele verschiedene Stile und Konfigurationen ausprobiert. Allerdings habe ich mich des Öfteren dabei ertappt, wie ich meine selbst erstellten Konfigurationen zu Gunsten des Metropolis-Stils [3] verworfen habe, welcher in meinen Augen der am besten gestaltete Beamer-Stil ist, den ich bisher gesehen habe.

Das größte Problem ist, dass Metropolis nicht mehr aktiv gepflegt wird. Das letzte Update war vor sechs Jahren und seitdem wurden zahlreiche Probleme gemeldet. Die meisten sind nicht weiter tragisch und können durch verschiedene Hacks und Behelfslösungen umgangen werden, allerdings wurde ich immer unzufriedener damit, eine zusätzliche Datei mit Fehlerkorrekturen für Metropolis vorhalten zu müssen, um diese Probleme zu beheben. Tatsächlich beinhaltet Beamer selbst auch mehrere Korrekturen, nur um den Stil am Leben zu halten.²³⁴

Moloch

Dieser Zustand hat mich nun (besser gesagt vor einigen Monaten) dazu veranlasst, einen Fork von Metropolis zu erstellen mit dem Ziel, all die offenen Probleme zu beheben. Ich nenne den neuen Stil Moloch, was allen Kennern von Metropolis vertraut sein dürfte. Das ursprüngliche Design ist weitestgehend erhalten, mit der Ausnahme von einigen kleineren Änderungen, die ich im Abschnitt »Änderungen«

¹ Ursprünglich veröffentlicht unter <https://jolars.co/blog/2024-05-30-moloch/>, übersetzt ins Deutsche von samcarter

² <https://github.com/josephwright/beamer/blob/c0d91f15165421646b5383546e6195187b7f97c9/base/beamerbasecompatibility.sty#L253>

³ <https://github.com/josephwright/beamer/blob/c0d91f15165421646b5383546e6195187b7f97c9/base/beamerbasesection.sty#L209>

⁴ <https://github.com/josephwright/beamer/blob/c0d91f15165421646b5383546e6195187b7f97c9/base/beamerbaselocalstructure.sty#L29>

zusammenfassen werde und die den Stil weiter an das gewohnte Verhalten von Beamer angleichen. Auch wurde der Quellcode des Stils vereinfacht und zuverlässiger gemacht. So nutzte Metropolis zum Beispiel `\patchcmd` aus dem `etoolbox`-Paket, um interne Beamer-Befehle zu verändern, etwa um damit Änderungen an Folientiteln vorzunehmen. Genau das hat zu Problemen geführt, als Beamer Änderungen an diesen Befehlen vorgenommen hat. Daher habe ich mich dafür entschieden, all diese Patches zu entfernen und stattdessen lieber auf die Bordmittel von Beamer zu bauen.

Dies hat allerdings seinen Preis. Einige Funktionen, wie das Umschalten des Titels zwischen Großbuchstaben, Kapitälchen und normalem Text, mussten dafür entfallen. Wie aber die Metropolis-Dokumentation selbst sagt,⁵ waren diese Modifikationen von Anfang an problematisch. Deshalb denke ich, dass das Entfernen dieser Funktionen richtig ist.⁶

Des Weiteren habe ich den `pgfplots`-Stil entfernt, der in Metropolis enthalten war. Ich habe nichts gegen diesen Stil, allerdings bin ich der Meinung, dass er nicht zu einem Allzweckstil passt.

Erste Schritte

Das Design des Stils ist eng an das Originaldesign von Metropolis angelehnt (und wird diesem auch in Zukunft treu bleiben). In Abbildung 1 ist ein kurzes Beispiel mit ein paar Folien im Stil von Moloch zu sehen.

Moloch ist auf CTAN verfügbar [1], man kann es also mittels des $\text{\TeX}$ Live-Paketmanagers installieren, indem man den folgenden Befehl ausführt:

```
tlmgr install moloch
```

Moloch ist in einem vollständigem $\text{\TeX}$ Live ab der Version 2024 enthalten und muss nicht extra installiert werden.

Die Nutzung des Stils ist so einfach wie bei jedem anderen, hier ein einfaches Beispiel:

```
\documentclass{beamer}

\usetheme{moloch}

\title{Title}
\author{Name}
```

⁵ siehe z.B. <https://github.com/matze/mtheme/issues/371>.

⁶ Anmerkung der Übersetzerin: Beamer-Nutzer können nun selbst das Titelformat festlegen, beispielsweise mit `\settitleformat`. Ironischerweise sind es ausgerechnet die Änderungen an den internen Befehlen, um dies zu ermöglichen, welche die Patches von Metropolis stören.

```
\institute{Institut}
\date{\today}

\begin{document}
  \maketitle
  \section{Erstes Kapitel}
  \begin{frame}
    \frametitle{Erste Folie}
    Hallo Welt!
  \end{frame}
\end{document}
```

Die Dokumentation des Stils beinhaltet detailliertere Informationen und eine Liste der verfügbaren Optionen.⁷ Für Nutzer von Metropolis ist am interessantesten, dass `\metroset` durch `\molochset` ersetzt wurde und dass einige Funktionen nicht mehr unterstützt werden. Welche genau das sind, werden wir im nächsten Abschnitt ausführlicher betrachten.

Änderungen

Im Folgenden versuche ich, eine Übersicht über die wichtigsten Änderungen zu geben.

Eine neue Sekundärfarbe

Ich war schon immer der Meinung, dass die grüne Farbe in Metropolis zu grell und nicht besonders gut für farbenblinde Personen geeignet ist. Daher habe ich sie durch ein dunkles Türkis ersetzt, das in meinen Augen etwas gedämpfter und angenehmer zu betrachten ist. Der Unterschied ist in Abbildung 2 zu sehen. Ich hoffe, dass die neue Farbe als Verbesserung betrachtet wird.

Untertitel

Moloch unterstützt nun auch Folien-Untertitel, was in Metropolis nicht der Fall war, da der Autor Untertitel für eine schlechte Idee hielt.⁸ Im Großen und Ganzen stimme ich ihm dabei zu, dass man auf Untertitel verzichten sollte, allerdings sehe ich keinen Grund, diese Meinung anderen aufzuzwingen. Darum werden Untertitel in Moloch unterstützt.

⁷<http://mirrors.ctan.org/macros/latex/contrib/beamer-contrib/themes/moloch/moloch.pdf>

⁸<https://github.com/matze/mtheme/issues/135>

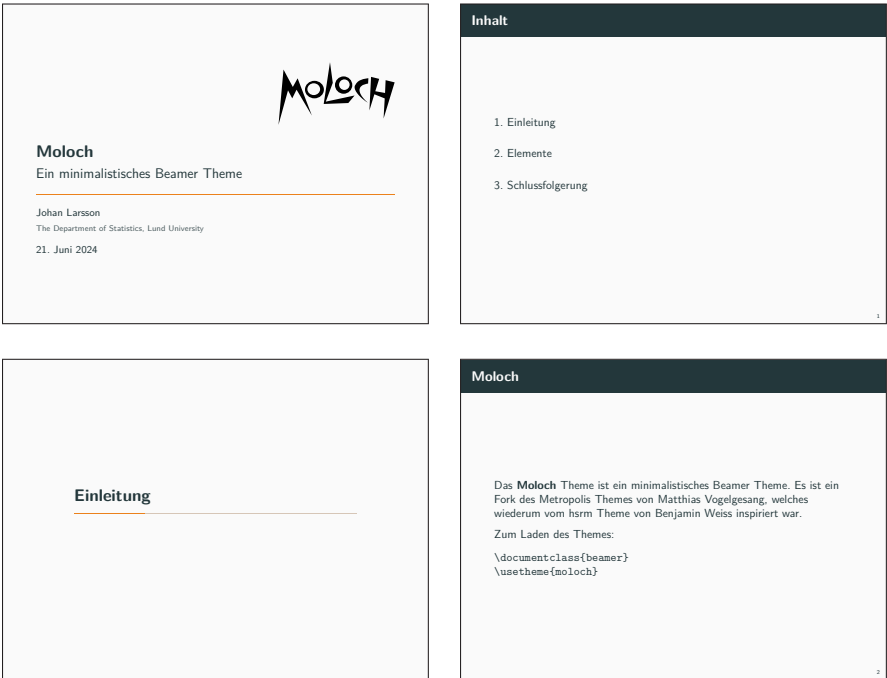


Abb. 1: Die ersten vier Folien einer Präsentationen mit Moloch.



Abb. 2: Vergleich zwischen der alten und neu-

en Sekundärfarbe.

Abb. 3: Folienuntertitel werden in Moloch un-

terstützt.

Folien-Nummerierung

Metropolis hatte sein eigenes System zur Nummerierung von Folien, woran an sich nichts auszusetzen ist. Um die Folien im Anhang separat zu zählen (und zu gewährleisten, dass sie nicht zur Gesamtzahl der Folien zählten), konnten Nutzer das Paket `appendixframenumber` verwenden. Beamer hat allerdings in den letzten Jahren sein eigenes System zur Nummerierung von Folien verbessert, sodass nun keine eigene Lösung oder separate Pakete mehr notwendig sind, um diese Funktionalität zu nutzen. Daher nutzt Moloch nur die Beamer-eigenen Lösungen zur Folien-Nummerierung. Deren Design ist allerdings ein klein wenig anders; so ist die Schriftgröße kleiner und die Ränder unterscheiden sich etwas, was in meinen Augen allerdings eine Verbesserung ist.

Nun kann man die Nummerierung mittels der gewohnten Beamer-Befehle ändern, zum Beispiel mittels:

```
\setbeamertemplate{page number in head/foot}[appendixframenumber]
```

Neugestaltung der Titelfolie

Die Titelfolie wurde neu gestaltet. Die wichtigsten Änderungen hierbei sind:

1. Das Institut ist nun direkt unterhalb des Autors platziert (anstelle des Datums), was, denke ich, mehr Sinn macht, da der Autor und das Institut enger zusammen hängen. Dies wurde in einem Metropolis-Github-Issue⁹ vorgeschlagen, aber nicht umgesetzt.
2. Ober- und unterhalb des Titelbildes gibt es nun etwas Freiraum. Zuvor war es als eine `\vbox` mit der Höhe Null gesetzt, sodass es das Layout der Titelfolie nicht beeinflusste. Nun wird es den Titel und sonstige Informationen auf der Titelseite entsprechend nach unten verschieben. Mit dem neuen Layout sind die Ränder oben und unten gleich groß und passen sich entsprechend der Größe des Titelbildes an. Falls dieses neue Verhalten nicht erwünscht ist, können die Nutzer ihr Titelbild selbst in eine `\vbox` mit einer Höhe von Null setzen. Dementsprechend betrachte ich das neue Verhalten als den weniger invasiven Ansatz.
3. Die Ränder um die einzelnen Elemente der Titelseite wurden komplett geändert, siehe Abbildung 4. Der Hauptunterschied ist, dass der Abstand zwischen Titel und Untertitel nun kleiner und der Freiraum um die orange Linie herum nun symmetrisch ist.

Ich bin offen für Vorschläge und Diskussionen, wie sich das Layout der Titelfolie weiter verbessern oder leichter an individuelle Bedürfnisse anpassen lässt.

⁹ <https://github.com/matze/mtheme/issues/180>



Abb. 4: Die Titleseite von Metropolis (links) und Moloch (rechts).

Schriftarten

Metropolis hat eine spezielle Herangehensweise bezüglich Schriftarten. Falls die Präsentation mit $\text{\LaTeX}$ oder $\text{\XeLaTeX}$ kompiliert wird, dann prüft Metropolis automatisch, ob die Schrift »Fira Sans Light« verfügbar ist und nutzt diese gegebenenfalls als Hauptschrift. Persönlich mag ich die Schrift »Fira Sans« und denke, dass sie eine gute Wahl für Präsentationen ist. Allerdings glaube ich nicht, dass sie ein Bestandteil eines Beamer-Stils sein sollte. Insbesondere glaube ich nicht, dass es wünschenswert ist, wenn sich die Resultate für $\text{\pdfLaTeX}$ standardmäßig von denen anderer Formate unterscheiden.

Daher habe ich die Einstellungen bezüglich Schriftarten entfernt. Nutzer, die das Aussehen von Metropolis auch bezüglich der Schriftart reproduzieren wollen, brauchen nur mit $\text{\XeLaTeX}$ oder $\text{\LaTeX}$ zu kompilieren und die Schrift ähnlich wie im folgen Beispiel zu ändern:

```
\usepackage{fontspec}

\setsansfont[
  ItalicFont={Fira Sans Light Italic},
  BoldFont={Fira Sans},
  BoldItalicFont={Fira Sans Italic}
]{Fira Sans Light}
\setmonofont[BoldFont={Fira Mono Medium}]{Fira Mono}

\AtBeginEnvironment{tabular}{%
  \addfontfeature{Numbers={Monospaced}}
}
```

Falls `\operatorname`, `\mathrm` etc. ebenfalls in Fira gesetzt werden sollen, muss auch `\setmainfont` entsprechend gesetzt werden.

Man beachte, dass von Fira Math Light nur eine Betaversion verfügbar ist, die bei Weitem nicht vollständig ist. Das bedeutet leider, dass es im Moment keine gute Lösung für eine zu Fira Sans Light passende Mathematikschrift gibt. Andernfalls könnte man das `unicode-math` Paket mit Fira Math Light benutzen. Meiner Meinung nach ist dies ein weiteres gutes Argument dafür, dass der Stil standardmäßig nicht die Fira-Schriften nutzen sollte.

Keine automatischen Abstände zwischen Absätzen

Im Gegensatz zu Beamer, in dem `\parskip` (entspricht in etwa dem Abstand zwischen zwei Absätzen) auf Null gesetzt wird, verwendet Metropolis hingegen einen Abstand von 0,5 em. Das hat zur Folge, dass in Metropolis kein `\medskip` oder ähnliches benötigt wird, um Absätze hübsch voneinander zu trennen.

Wie ich in einigen Moloch-Github-Issues angemerkt habe,¹⁰¹¹¹² hat dies einige unerwünschte Nebeneffekte, wie beispielsweise zusätzlichen Abstand zwischen Tabellenüberschriften und Tabellen. Demzufolge habe ich diese Einstellungen aus Moloch entfernt.

Wie auch viele der anderen Änderungen, nähert dies Moloch weiter an das gewohnte Verhalten von Beamer an. Dies ist in meinen Augen gut, da es den Wechsel zwischen verschiedenen Stilen vereinfacht.

Block-Umgebungen

Metropolis hat einen eigenen Ansatz für die Handhabung von Block-Umgebungen. Insbesondere die Blöcke mit gefülltem Hintergrund wurden verändert, sodass der Text außerhalb des Blocks bündig mit dem Rahmen des Blocks ist und nicht mit dem Text im Inneren des Blocks, wie es in Beamer üblich ist; siehe Abbildung 5 für einen Vergleich.

Ich denke, dass das übliche Verhalten von Beamer die richtige Wahl ist. Insbesondere, da ansonsten der Inhalt der Blöcke anders gesetzt wird, wenn man zu Blöcken mit gefülltem Hintergrund wechselt. Hinzu kommt, dass die Abstände in Metropolis-Blöcken nicht richtig funktionieren,¹³ was durch den Wechsel zu normalen Beamer-Blöcken ebenfalls behoben wird.

¹⁰ <https://github.com/jolars/moloch/issues/8>

¹¹ <https://github.com/jolars/moloch/issues/9>

¹² siehe auch den Beamer-Github-Issue <https://github.com/josephwright/beamer/issues/512> für zusätzliche Hintergrundinformationen

¹³ siehe <https://github.com/matze/mtheme/issues/307>

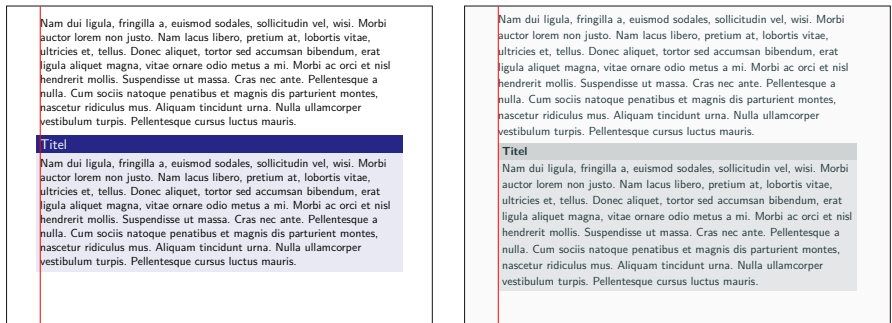


Abb. 5: Ausrichtung der Block-Umgebung in Beamer (links) und Metropolis (rechts).

Build-System

Moloch ist auf CTAN verfügbar und in $\text{\TeX}$ Live 2024 enthalten. Im Normalfall muss man sich um die Installation also keine Gedanken machen. Für den Fall, dass ein Nutzer Moloch manuell installieren möchte, beispielsweise um neue Features oder Fehlerbehebungen zu nutzen, die in der Entwicklungsversion enthalten sind, nutzt Moloch anstelle eines Makefiles nun `l3build` [2], um das Paket zu bauen. Das sollte für die meisten Nutzer das Leben einfacher machen, da man nur noch die folgenden Befehle aufrufen muss:

```
git clone https://github.com/jolars/moloch.git
cd moloch
l3build install
```

Darüber hinaus bedeutet dies auch, dass das Paket nun eigene Testfälle enthält, um sicherzustellen, dass es zu keinen unerwarteten Problemen mit dem Stil kommt.

Sonstige Änderungen

Es gibt weitere kleinere Änderungen, von denen ich im Folgenden einige aufliste:

- In Breitbild-Präsentationen funktionieren die Fortschrittsbalken nun besser¹⁴.
- Änderungen an Kapiteln im Anhang wurden rückgängig gemacht¹⁵.
- Das Aussehen der Bibliografie ist nun nicht mehr fest vorgegeben¹⁶.

¹⁴ <https://github.com/matze/mtheme/pull/384>

¹⁵ <https://github.com/matze/mtheme/pull/342>

¹⁶ <https://github.com/jolars/moloch/issues/5>

Zukünftige Pläne

Ich plane im Moment keine weiteren drastischen Änderungen und werde den Stil wahrscheinlich bald als stabile Version veröffentlichen, sodass sich die Nutzer darauf verlassen können, dass der Stil keine inkompatiblen Änderungen mehr bekommt. Ich denke, dass das Originaldesign von Metropolis großartig ist und möchte mich nicht zu weit davon entfernen.

Des Weiteren möchte ich die Farben im Stil einfacher anpassbar machen und eventuell alternative Farbschemata hinzufügen. Das bedeutet auch, das kontrastreiche Alternativ-Farbschema aus Metropolis wiedereinzuführen, welches ich aus Moloch entfernt habe (aus Gründen, die mir im Moment entfallen sind). Wie dem auch sei, ich plane, die Standardeinstellungen nicht weiter zu verändern.

Unterstützung

Alle, die denken, dass sie etwas zum Projekt beitragen können, bitte gerne! Das Projekt ist bei GitHub gehostet und Nutzer können dort Probleme melden oder Diskussionen anstoßen, falls es etwas gibt, was verbessert werden kann.¹⁷

Danksagung

samcarter hat mit Diskussionen und Tests des Themes geholfen und auch dazu beigetragen, den Übergang von Metropolis zu Moloch reibungsloser zu gestalten.

Zum Schluss: Ehre, wem Ehre gebührt. Ich möchte betonen, dass der Großteil des Moloch-Codes von Matthias Vogelsang geschrieben wurde, dem Autor von Metropolis und dass meine Aufgabe hauptsächlich darin bestand, die Schwachstellen zu glätten.

Literatur

- [1] Johan Larsson: The moloch package, Beamer based on Metropolis theme, Version 0.4.0, 2024, <https://ctan.org/pkg/moloch>.
- [2] The LaTeX Project Team: The l3build package, A testing and building system for (La)TeX, 2024, <https://ctan.org/pkg/l3build>.
- [3] Matthias Vogelgesang: The beamertheme-metropolis package, A modern $\LaTeX$ beamer theme, Version 1.2, 2017, <https://ctan.org/pkg/beamertheme-metropolis>.

¹⁷ <https://github.com/jolars/moloch>

JetBrains Mono

Herbert Voß

Einführung



Die Schreibmaschinenschrift JetBrains Mono von Philipp Nurullin und Konstantin Bulenkov (<https://www.jetbrains.com/de-de/lp/mono/#font-family>) wurde schon sehr kurz in Niepraschk [2] vorgestellt. Dabei wurde nicht extra erwähnt, dass sie über Ligaturen verfügt, mit denen Pfeile oder pfeilähnliche Symbole dargestellt werden können. Darauf soll hier nun näher eingegangen werden. Diese Ligaturen erlauben die Benutzung innerhalb von Listings, ohne dass auf speziellen Code zurückgegriffen werden muss. Möchte man beispielsweise einen Pfeil darstellen, so erfolgt dies mit der einfachen Eingabe von `->`, was zur Ausgabe `→` führt, da hierfür in der Schrift eine Ligatur existiert, die beide Eingabezeichen zu einem Zeichen verbindet.

Eingabe	Ausgabe	Eingabe	Ausgabe	Eingabe	Ausgabe
<code><></code>	<code>◊</code>	<code> =></code>	<code>⇨</code>	<code><= </code>	<code>⇐=</code>
<code>++</code>	<code>⦿</code>	<code>-></code>	<code>→</code>	<code><-</code>	<code>←</code>
<code><-></code>	<code>↔</code>	<code><! --</code>	<code>↔!—</code>	<code>~></code>	<code>↗</code>
<code><~</code>	<code>↖</code>	<code>~~></code>	<code>↗</code>	<code><~~</code>	<code>↖</code>
<code>->></code>	<code>→</code>	<code><<-</code>	<code>←</code>	<code>--></code>	<code>→</code>
<code><--</code>	<code>←</code>	<code>>-></code>	<code>→</code>	<code><-<</code>	<code>←</code>
<code>/=</code>	<code>⊢</code>	<code>=></code>	<code>⇒</code>	<code><=</code>	<code>≤</code>
<code><=></code>	<code>⇔</code>	<code>=>></code>	<code>⇒</code>	<code><=<</code>	<code>⇐</code>
<code>==></code>	<code>⇒</code>	<code><==</code>	<code>⇐</code>	<code><==></code>	<code>⇔</code>
<code>>=></code>	<code>⇒</code>	<code><=<</code>	<code>⇐</code>	<code>\#\#</code>	<code>##</code>
<code>\#\#\#</code>	<code>###</code>	<code> ></code>	<code>▷</code>	<code>< </code>	<code>◁</code>
<code>[]</code>	<code>[]</code>	<code>:?></code>	<code>:?></code>	<code>::=</code>	<code>::=</code>
<code>::</code>	<code>::</code>	<code>??</code>	<code>??</code>	<code><+></code>	<code><+></code>
<code>..</code>	<code>..</code>	<code></</code>	<code></</code>	<code>/></code>	<code>/></code>
<code>\#?</code>	<code>#?</code>	<code>]\#</code>	<code>]#</code>		

Wie man obiger Tabelle entnehmen kann, sollte die Eingabe von `~~>` zum Photonen-symbol führen, es kommt jedoch nur `→` als Ausgabe. Damit die Schrift die Ligaturen eindeutig erkennt, dürfen die Zeichen nicht aktiv sein, was für $\text{\TeX}$ bei mehreren

Zeichen der Fall ist. Die Tilde ist bekanntermaßen das sogenannte nobreakspace, definiert als `\active@prefix ~\active@char~`. In solchen Fällen muss ein Befehl verwendet werden, der eindeutig nur ein Zeichen darstellt. In diesem Fall ist es `\textasciitilde`, womit dann `{\JetBrains\textasciitilde\textasciitilde}` zu `~~~` führt.

Die Auswahl der TrueType- oder OpenType-Schrift kann auch global als Schreibmaschinenschrift erfolgen:

```
\usepackage{fontspec}
\setmonofont{JetBrains Mono}[Renderer=HarfBuzz]
```

Oder als spezieller Befehl:

```
\usepackage{fontspec}
\newfontface\JetBrains{JetBrains Mono}[Renderer=HarfBuzz]
```

Wird im aktuellen Dokument auch der Verbatimmodus verwendet, so müssen für diesen die Ligaturen abgeschaltet werden, was in der Präambel wie folgt möglich ist:

```
\makeatletter
\AddToHook{cmd/verbatim@nolig@list/after}{\do ||\do |=\do +\do ~\do \#
\do \.\do \/ \do \?\do \:\do \[\do \]}
\makeatother
```

Schriftdateien

Standardversionen

Auf der offiziellen Webseite von JetBrains gibt es ausschließlich TrueType-Varianten der Schrift zum Download, wobei die JetBrainsMonoNL-Dateien über keine Ligaturen verfügen:

JetBrainsMono-Bold.ttf	JetBrainsMono-BoldItalic.ttf
JetBrainsMono-ExtraBold.ttf	JetBrainsMono-ExtraBoldItalic.ttf
JetBrainsMono-ExtraLight.ttf	JetBrainsMono-ExtraLightItalic.ttf
JetBrainsMono-Italic.ttf	JetBrainsMono-Light.ttf
JetBrainsMono-LightItalic.ttf	JetBrainsMono-Medium.ttf
JetBrainsMono-MediumItalic.ttf	JetBrainsMono-Regular.ttf
JetBrainsMono-SemiBold.ttf	JetBrainsMono-SemiBoldItalic.ttf
JetBrainsMono-Thin.ttf	JetBrainsMono-ThinItalic.ttf
JetBrainsMonoNL-Bold.ttf	JetBrainsMonoNL-BoldItalic.ttf
JetBrainsMonoNL-ExtraBold.ttf	JetBrainsMonoNL-ExtraBoldItalic.ttf
JetBrainsMonoNL-ExtraLight.ttf	JetBrainsMonoNL-ExtraLightItalic.ttf
JetBrainsMonoNL-Italic.ttf	JetBrainsMonoNL-Light.ttf
JetBrainsMonoNL-LightItalic.ttf	JetBrainsMonoNL-Medium.ttf
JetBrainsMonoNL-MediumItalic.ttf	JetBrainsMonoNL-Regular.ttf

JetBrainsMonoNL-SemiBold.ttf
 JetBrainsMonoNL-Thin.ttf

JetBrainsMonoNL-SemiBoldItalic.ttf
 JetBrainsMonoNL-ThinItalic.ttf

OpenType-Versionen

Auf der Entwicklerseite <https://github.com/JetBrains/JetBrainsMono> gibt es die Schriften auch in einer OpenType-Variante, die hier im Folgenden in der Version 2.305 benutzt werden.

JetBrainsMono-Bold.otf	JetBrainsMono-BoldItalic.otf
JetBrainsMono-ExtraBold.otf	JetBrainsMono-ExtraBoldItalic.otf
JetBrainsMono-ExtraLight.otf	JetBrainsMono-ExtraLightItalic.otf
JetBrainsMono-Italic.otf	JetBrainsMono-Light.otf
JetBrainsMono-LightItalic.otf	JetBrainsMono-Medium.otf
JetBrainsMono-MediumItalic.otf	JetBrainsMono-Regular.otf
JetBrainsMono-SemiBold.otf	JetBrainsMono-SemiBoldItalic.otf
JetBrainsMono-Thin.otf	JetBrainsMono-ThinItalic.otf

Die verschiedenen Schriftschnitte zeigt die folgende Zusammenstellung:

JetBrainsMono-Thin.otf	JetBrainsMono-ExtraLight.otf
JetBrainsMono-Light.otf	JetBrainsMono-Regular.otf
JetBrainsMono-Medium.otf	JetBrainsMono-SemiBold.otf
JetBrainsMono-ExtraBold.otf	
JetBrainsMono-ExtraLightItalic.otf	JetBrainsMono-ThinItalic.otf
JetBrainsMono-LightItalic.otf	JetBrainsMono-Italic.otf
JetBrainsMono-MediumItalic.otf	JetBrainsMono-SemiBoldItalic.otf
JetBrainsMono-BoldItalic.otf	JetBrainsMono-ExtraBoldItalic.otf

Version mit variabler Schriftwichtung

Es stehen auch zwei variable Fonts¹ in einer TrueType-Version zur Verfügung:

JetBrainsMono-Italic[wght].ttf
 JetBrainsMono[wght].ttf

Diese können über die Option `+axis` auf eine bestimmte Wichtung gesetzt werden, wobei die Zuweisung über die `fontspec`-Option `RawFeature` erfolgen muss. Beispielsweise:

```
\setmonofont{JetBrainsMono[wght].ttf}[Scale=0.8,FakeStretch=0.9,  
  RawFeature={+axis={wght=400}}]% Standardwert
```

¹ Siehe <https://v-fonts.com>, <https://fonts.google.com/variablefonts?query=Jet&vfquery=JetBra> und für die Definition von variablen Schriften auch Beinert [1]

Die Wichtung ist standardmäßig auf 400 gesetzt und kann in Einerstufen im Bereich von 100 bis 800 geändert werden.

JetBrains Mono variable font 100	<i>JetBrains Mono-Italic variable font 100</i>
JetBrains Mono variable font 200	<i>JetBrains Mono-Italic variable font 200</i>
JetBrains Mono variable font 300	<i>JetBrains Mono-Italic variable font 300</i>
JetBrains Mono variable font 400	<i>JetBrains Mono-Italic variable font 400</i>
JetBrains Mono variable font 500	<i>JetBrains Mono-Italic variable font 500</i>
JetBrains Mono variable font 600	<i>JetBrains Mono-Italic variable font 600</i>
JetBrains Mono variable font 700	<i>JetBrains Mono-Italic variable font 700</i>
JetBrains Mono variable font 800	<i>JetBrains Mono-Italic variable font 800</i>

Eine variable Zeichenbreite wird von der JetBrainsMono nicht unterstützt.

Installation der Schrift

Da die oben angegebenen TrueType- oder OpenType-Dateien ohnehin nur mit $\text{X}_{\text{E}}\text{TeX}$ oder LuaTeX direkt genutzt werden können, empfiehlt sich eine Installation unabhängig von der vorhandenen TeX -Installation:

Linux Speichern in `/usr/local/share/`;

macOS speichern in `/Users/.../Library/` oder `/Library/`, falls die Dateien für alle User zur Verfügung stehen sollen;

Windows speichern in `C:\Windows\Fonts`.

Beispiel für Programmcode mit JetBrains Mono

```
\documentclass[ngerman]{dtk}
\usepackage{booktabs}
\newfontface\JetBrains{JetBrains Mono}
  [Renderer=HarfBuzz,Scale=0.8,FakeStretch=0.9]
\setmonofont{JetBrains Mono}[Renderer=HarfBuzz,Scale=0.8,FakeStretch=0.9]
\usepackage{unicodefonttable}
\usepackage{longtable}
\makeatletter
\AddToHook{cmd/verbatim@nolig@list/after}{%
  \do ||\do \=\do \+\do \~\do \#\do \.\do \/\/\do \?\/\do \:\do \[\do \]}
\makeatother

\title{JetBrains Mono} %Pfeile ohne Mathematikmodus}
\Author{Herbert}{Voß}{\Email{Herbert.Voss@FU-Berlin.de}}
\begin{document}
\maketitle
```

Verfügbare Schriftzeichen

Grundsätzlich kann die Schrift auch generell als Monoschrift-Variante gewählt werden. Sie verfügt über die folgenden Zeichen:

Tab. 2: JetBrainsMono-Regular.otf																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Basic Latin																
U+0000 - 000F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
U+0020 - 002F		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
U+0030 - 003F	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
U+0040 - 004F	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
U+0050 - 005F	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
U+0060 - 006F	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
U+0070 - 007F	p	q	r	s	t	u	v	w	x	y	z	{		}	~	-
Latin-1 Supplement																
U+00A0 - 00AF		ı	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	­	®	¯
U+00B0 - 00BF	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
U+00C0 - 00CF	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
U+00D0 - 00DF	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
U+00E0 - 00EF	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
U+00F0 - 00FF	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ
Latin Extended-A																
U+0100 - 010F	Ā	ā	Ă	ă	Ą	ą	Ć	ć	Ĉ	ĉ	Ċ	ċ	Č	č	Ď	ď
U+0110 - 011F	Ē	ē	Ĕ	ĕ	Ė	ė	Ę	ę	Ě	ě	Ĝ	ĝ	Ğ	ğ	Ĥ	ĥ
U+0120 - 012F	Ġ	ġ	Ģ	ģ	Ĥ	ĥ	Ħ	ħ	Ĩ	ĩ	Ĭ	ĭ	Į	į	İ	ı
U+0130 - 013F	Ĳ	ķ	Ļ	ļ	Ņ	ņ	Ņ	ņ	Ň	ň	Ŋ	ŋ	Ō	ō	Ŏ	ö
U+0140 - 014F	Ű	ű	Ų	ų	Ŵ	ŵ	Ŷ	ŷ	Ÿ	ž	Ž	ž	Ž	ž	Ž	ž
U+0150 - 015F	Š	š	Ţ	ţ	Ť	ť	Ŧ	ŧ	Ũ	ũ	Ū	ū	Ŭ	ŭ	Ů	ů
U+0160 - 016F	Ų	ų	Ŵ	ŵ	Ŷ	ŷ	Ÿ	ž	Ž	ž	Ž	ž	Ž	ž	Ž	ž
U+0170 - 017F	Ų	ų	Ŵ	ŵ	Ŷ	ŷ	Ÿ	ž	Ž	ž	Ž	ž	Ž	ž	Ž	ž
Latin Extended-B																
U+0180 - 018F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ø
U+0190 - 019F	Ɔ	-	ƒ	-	-	-	-	-	-	-	-	Ƶ	-	-	-	-
U+01A0 - 01AF	Œ	œ	-	-	-	-	-	-	-	-	-	-	-	-	-	Ű

Tab. 2: JetBrainsMono-Regular.otf *cont.*

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+0180 - 01BF	Ů	-	-	-	-	-	-	-	-	-	-	-	-	Ǻ	-	-
U+01C0 - 01CF	-	-	-	-	-	-	-	-	-	-	-	-	-	Ǻ̃	ǻ	-
U+01D0 - 01DF	Ǫ	-	ǫ	-	ǫ̃	-	ǫ̃	-	ǫ̃	-	ǫ̃	-	ǫ̃	-	-	-
U+01E0 - 01EF	-	-	-	-	-	-	Ǫ̃	ǫ̃	-	-	Ǫ̃	ǫ̃	-	-	-	-
U+01F0 - 01FF	-	-	-	-	Ǫ̃	ǫ̃	-	-	-	-	-	-	Ǫ̃	ǫ̃	Ǫ̃	ǫ̃
U+0210 - 021F	-	-	-	-	-	-	-	-	Œ	ŷ	Ŧ	ŧ	-	-	-	-
U+0230 - 023F	-	-	Ȳ	ȳ	-	-	-	Ȳ	-	-	-	-	-	-	-	-
IPA Extensions																
U+0250 - 025F	-	-	-	-	-	-	-	-	-	ə	-	-	-	-	-	-
Spacing Modifier Letters																
U+02B0 - 02BF	-	-	-	-	-	-	-	-	-	ˆ	ˆ	-	ˆ	-	-	-
U+02C0 - 02CF	-	-	-	-	-	-	-	-	-	ˆ	ˆ	-	ˆ	-	-	-
U+02D0 - 02DF	-	-	-	-	-	-	-	-	-	ˆ	ˆ	-	ˆ	-	-	-
U+02F0 - 02FF	-	-	-	-	-	-	-	-	-	ˆ	ˆ	-	ˆ	-	-	-
Combining Diacritical Marks																
U+0300 - 030F	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ
U+0310 - 031F	-	-	ˆ	-	-	-	-	-	-	-	-	ˆ	-	-	-	-
U+0320 - 032F	-	-	-	ˆ	-	ˆ	ˆ	ˆ	ˆ	-	-	-	-	-	-	-
U+0330 - 033F	-	-	-	-	-	ˆ	ˆ	ˆ	ˆ	-	-	-	-	-	-	-
Greek and Coptic																
U+0370 - 037F	-	-	-	-	ˆ	ˆ	-	-	-	-	-	-	-	ˆ	ˆ	-
U+0380 - 038F	-	-	-	-	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ	ˆ
U+0390 - 039F	ˆ	Α	Β	Γ	Δ	Ε	Ζ	Η	Θ	Ι	Κ	Λ	Μ	Ν	Ξ	Ο
U+03A0 - 03AF	Π	Ρ	-	Σ	Τ	Υ	Φ	Χ	Ψ	Ω	Ͱ	ͱ	Ͳ	ͳ	ʹ	͵
U+03B0 - 03BF	Ͷ	α	β	γ	δ	ε	ζ	η	θ	ι	κ	λ	μ	ν	ξ	ο
U+03C0 - 03CF	π	ρ	ς	σ	τ	υ	φ	χ	ψ	ω	Ͱ	ͱ	Ͳ	ͳ	ʹ	͵
U+03D0 - 03DF	-	-	-	-	-	ϕ	ϖ	ϗ	-	-	-	-	-	-	-	-
Cyrillic																
U+0400 - 040F	-	Ё	Ђ	Ѓ	Є	Ѕ	І	Ї	Ј	Љ	Њ	Ћ	Ќ	-	Ў	Ѓ
U+0410 - 041F	А	Б	В	Г	Д	Е	Ж	З	И	Й	К	Л	М	Н	О	П

Tab. 2: JetBrainsMono-Regular.otf *cont.*

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+0420 - 042F	Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я
U+0430 - 043F	а	б	в	г	д	е	ж	з	и	й	к	л	м	н	о	п
U+0440 - 044F	р	с	т	у	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я
U+0450 - 045F	-	ё	ђ	ѓ	є	ѕ	і	ї	ј	љ	њ	ћ	ќ	-	ѣ	ѵ
U+0490 - 049F	Ґ	ґ	Ғ	ғ	-	-	-	-	-	-	Қ	қ	-	-	-	-
U+04A0 - 04AF	-	-	Ң	ң	-	-	-	-	-	-	-	-	-	-	Ү	ү
U+04B0 - 04BF	Ұ	ұ	-	-	-	-	Ҷ	ҷ	-	-	Һ	һ	-	-	-	-
U+04D0 - 04DF	-	-	-	-	-	-	-	-	Ә	ә	-	-	Ӓ	ӓ	Ӥ	ӥ
U+04E0 - 04EF	-	-	-	-	Ӧ	ӧ	Ӱ	ӱ	Ө	ө	-	-	-	-	-	-
U+04F0 - 04FF	-	-	-	-	Ӣ	ӣ	-	-	-	-	-	-	-	-	-	-

Gujarati

U+0AE0 - 0AEF	-	-	-	-	-	-	-	-	-	-	ઁ	-	-	-	-	-
---------------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Latin Extended Additional

U+1E80 - 1E8F	Ẁ	ẁ	Ẃ	ẃ	Ẅ	ẅ	-	-	-	-	-	-	-	-	-	-
U+1E90 - 1E9F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ẞ	-
U+1EA0 - 1EAF	Ạ	ạ	Ả	ả	Ẳ	ẳ	Ẵ	ẵ	Ẹ	ẹ	Ẽ	ẽ	Ẹ̇	ẹ̇	Ẹ̈	ẹ̈
U+1EB0 - 1EBF	Ẹ̊	ẹ̊	Ẹ̋	ẹ̋	Ẹ̌	ẹ̌	Ẹ̍	ẹ̍	Ẹ̎	ẹ̎	Ẹ̏	ẹ̏	Ẹ̐	ẹ̐	Ẹ̑	ẹ̑
U+1EC0 - 1ECF	Ẹ̒	ẹ̒	Ẹ̓	ẹ̓	Ẹ̔	ẹ̔	Ẹ̕	ẹ̕	Ẹ̖	ẹ̖	Ẹ̗	ẹ̗	Ẹ̘	ẹ̘	Ẹ̙	ẹ̙
U+1ED0 - 1EDF	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ
U+1EE0 - 1EEF	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ
U+1EF0 - 1EFF	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	Ỗ	ỗ	-	-	-	-	-	-

General Punctuation

U+2000 - 200F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
U+2010 - 201F	-	-	-	-	-	-		-	‘	’	,	‘	”	”	”	”
U+2020 - 202F	†	‡	•	-	•	-	…	-	-	-	-	-	-	-	-	-
U+2030 - 203F	%	-	’	”	”	-	-	-	-	<	>	-	!!	?	-	„
U+2040 - 204F	-	-	-	-	/	[	]	-	-	-	-	-	-	-	-	-

Superscripts and Subscripts

U+2070 - 207F	0	-	-	-	4	5	6	7	8	9	+	-	=	-	-	-
U+2080 - 208F	0	1	2	3	4	5	6	7	8	9	-	-	-	-	-	-

Tab. 2: JetBrainsMono-Regular.otf cont.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Currency Symbols																
U+20A0 - 20AF	-	-	-	-	-	-	-	-	-	-	-	₠	€	-	₣	-
U+20B0 - 20BF	-	-	-	-	¢	-	-	-	-	-	-	-	-	₤	-	฿
Letterlike Symbols																
U+2100 - 210F	-	-	©	-	-	-	-	-	-	-	-	-	-	℥	-	-
U+2110 - 211F	-	-	-	ℓ	-	N	Nº	-	-	P	Q	-	-	R	-	-
U+2120 - 212F	-	-	™	-	Z	-	-	-	-	-	-	-	-	-	e	-
U+2140 - 214F	Σ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Number Forms																
U+2180 - 218F	-	-	-	-	-	-	-	-	-	-	Ⅎ	£	-	-	-	-
Arrows																
U+2190 - 219F	←	↑	→	↓	↔	↕	↖	↗	↘	↙	-	-	-	↻	↩	-
U+21A0 - 21AF	⇒	-	⇐	⇒	-	⇑	⇔	⇓	-	↺	↻	-	-	↺	-	-
U+21B0 - 21BF	-	-	-	-	-	-	-	-	-	-	-	-	-	-	↑	-
U+21C0 - 21CF	-	-	-	-	-	-	-	-	-	↩	-	-	-	-	-	-
U+21D0 - 21DF	⇐	⇑	⇒	⇓	⇔	-	-	-	-	-	-	⇒	-	-	⇕	⇕
U+21E0 - 21EF	-	-	-	-	-	⇨	-	⇑	⇨	-	⇕	-	-	-	-	-
Mathematical Operators																
U+2200 - 220F	∀	∃	∂	∃	∂	∅	Δ	∇	∈	∉	∈	∋	∉	-	■	∏
U+2210 - 221F	∏	Σ	-	∓	-	/	-	°	•	√	-	-	-	-	∞	-
U+2220 - 222F	-	-	-		ℓ	∥	-	∧	∨	∩	∪	∫	-	-	-	-
U+2230 - 223F	-	-	-	-	∴	∴	:	∴	÷	÷	-	-	≈	-	≈	-
U+2240 - 224F	-	-	-	≈	≈	-	-	≈	≈	≈	-	≈	-	≈	-	-
U+2250 - 225F	-	-	-	-	∴	-	-	∴	-	-	-	-	-	≈	-	≈
U+2260 - 226F	≠	≡	≠	≡	≤	≥	-	-	-	-	≪	≫	≈	≈	≈	≈
U+2270 - 227F	≠	≠	≤	≥	-	-	-	-	-	-	≪	≫	≈	≈	≈	≈
U+2280 - 228F	-	-	⊂	⊃	⊄	⊅	⊆	⊇	⊈	⊉	-	⊗	⊗	⊗	⊗	⊗
U+2290 - 229F	⊂	⊆	⊆	⊂	⊂	⊕	⊖	⊗	⊗	⊗	-	⊗	⊗	⊗	⊗	⊗
U+22A0 - 22AF	⊗	⊗	⊗	⊗	⊗	⊗	-	-	-	-	-	√	√	√	-	-
U+22B0 - 22BF	-	-	-	-	√	-	-	√	-	-	-	√	√	√	-	-
U+22C0 - 22CF	-	-	∩	∪	∘	-	*	-	⊗	⊗	⊗	-	-	-	Y	-
U+22D0 - 22DF	⊆	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
U+22E0 - 22EF	-	-	≠	-	-	-	-	-	-	-	-	-	-	-	⋮	⋮

Tab. 2: JetBrainsMono-Regular.ottf *cont.*

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+22F0 - 22FF	⋯	⋯	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Miscellaneous Technical

U+2300 - 230F	-	-	␣	^	⋈	-	-	⌈	⌋	⌌	⌍	-	-	-	-	-
U+2310 - 231F	-	-	-	-	-	-	-	⌘	-	-	-	⌎	⌏	⌐	⌑	⌒
U+2320 - 232F	-	-	-	-	⋈	⌞	⊠	⌠	-	⌡	-	⊠	-	-	-	-
U+2330 - 233F	-	-	-	-	-	-	⌢	⌣	⌤	⌥	⌦	⌧	⌨	〈	〉	⌫
U+2340 - 234F	⌬	⌭	⌮	⌯	⌰	⌱	⌲	⌳	⌴	⌵	⌶	⌷	⌸	⌹	⌺	⌻
U+2350 - 235F	⌼	⌽	⌿	⌻	⌽	⌿	⌽	⌿	⌽	⌿	⌽	⌿	⌽	⌿	⌽	⌿
U+2360 - 236F	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿
U+2370 - 237F	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿
U+2380 - 238F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
U+2390 - 239F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
U+23A0 - 23AF	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿
U+23C0 - 23CF	-	-	-	-	-	-	-	-	-	-	-	-	-	-	⌿	-
U+23F0 - 23FF	-	-	-	-	-	-	-	-	-	-	-	⌿	⌿	⌿	⌿	-

Control Pictures

U+2400 - 240F	N _U	S _H	S _X	E _X	E _T	E _Q	A _K	B _L	B _S	H _T	L _F	V _T	F _F	C _R	S _O	S _I
U+2410 - 241F	D _E	D ₁	D ₂	D ₃	D ₄	N _K	S _N	E _B	C _N	E _M	S _B	E _C	F _S	G _S	R _S	U _S
U+2420 - 242F	S _P	D _L	-	-	N _L	-	-	-	-	-	-	-	-	-	-	-

Box Drawing

U+2500 - 250F	─	─	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃
U+2510 - 251F	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃
U+2520 - 252F	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃
U+2530 - 253F	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃
U+2540 - 254F	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃
U+2550 - 255F	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃
U+2560 - 256F	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃
U+2570 - 257F	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃	┃

Block Elements

U+2580 - 258F	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
---------------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Tab. 2: JetBrainsMono-Regular.otf cont.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+2590 - 259F	▀	▄	▒	▓	▬		▩	▨	▧	▦	▥	▤	▣	▢	□	■
Geometric Shapes																
U+25A0 - 25AF	■	□	-	-	-	-	-	-	-	-	▪	▫	-	-	-	-
U+25B0 - 25BF	-	-	▲	△	▴	▵	▶	▷	►	▸	▹	►	▻	▼	▽	▾
U+25C0 - 25CF	◀	◁	◄	◅	◄	◅	◆	◇	◈	◎	◊	◯	⦶	-	⊙	●
U+25D0 - 25DF	-	-	-	-	◐	◑	-	-	-	-	-	-	-	-	-	-
U+25E0 - 25EF	-	-	-	-	-	-	◦	◼	◻	◻	◻	◻	◻	-	-	◯
U+25F0 - 25FF	-	-	-	-	-	-	Ⓒ	-	-	-	-	-	-	-	-	-
Miscellaneous Shapes																
U+2660 - 266F	-	-	-	-	-	-	-	-	-	-	-	-	-	♠	-	#
U+2680 - 268F	-	-	-	-	-	-	-	☺	-	-	-	-	-	-	-	-
U+26A0 - 26AF	⚠	⚡	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Dingbats																
U+2710 - 271F	-	-	-	✓	-	✕	-	✕	-	-	-	-	-	-	-	-
U+2730 - 273F	-	-	-	-	-	-	*	-	-	-	-	-	-	-	-	-
U+2760 - 276F	-	-	-	-	-	-	-	-	-	-	-	-	-	-	⌞	⌟
U+2770 - 277F	⌠	⌡	-	-	-	-	-	-	-	-	-	-	-	-	-	-
U+2790 - 279F	-	-	-	-	→	-	-	-	-	-	-	-	→	→	→	-
Miscellaneous Mathematical Symbols-A																
U+27C0 - 27CF	-	-	-	-	-	℘	ℙ	-	-	-	-	-	-	-	-	-
U+27D0 - 27DF	-	-	-	-	-	-	-	-	-	-	-	-	⌵	-	-	-
U+27E0 - 27EF	-	-	-	-	-	-	⌜	⌝	⌞	⌟	⌠	⌡	-	-	-	-
Supplemental Arrows-A																
U+27F0 - 27FF	-	-	-	-	-	←	→	↔	-	-	-	-	-	-	-	-
Supplemental Arrows-B																
U+2910 - 291F	-	-	-	-	-	↘	-	-	-	-	-	-	-	-	-	-
U+2940 - 294F	-	-	-	-	-	-	-	-	-	-	↙	-	-	-	-	-

Tab. 2: JetBrainsMono-Regular.otf *cont.*

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Miscellaneous Mathematical Symbols-B																
U+2980 - 298F	-	-	-	-	-	-	-	⌚	⌚	-	-	-	-	-	-	-
Supplemental Mathematical Operators																
U+2A00 - 2A0F	⊙	-	-	-	-	∏	∏	-	-	-	-	-	-	-	-	-
Miscellaneous Symbols and Arrows																
U+2B50 - 2B5F	-	-	-	-	-	-	-	-	⊙	-	-	-	-	-	-	-
Private Use Area																
U+E0A0 - E0AF	↯	ℒ	ℒ	-	-	-	-	-	-	-	-	-	-	-	-	-
U+E0B0 - E0BF	▶	>	◀	<	-	-	-	-	-	-	-	-	-	-	-	-
Small Form Variants																
U+FE60 - FE6F	-	-	+	-	-	-	-	-	-	-	-	-	-	-	-	-
Arabic Presentation Forms-B																
U+FEF0 - FEFF	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Halfwidth andFullwidth Forms																
U+FF50 - FF5F	-	-	-	-	-	-	-	-	-	-	-	{	-	}	-	-
Specials ...																
U+FFF0 - FFFF	-	-	-	-	-	-	-	-	-	-	-	-	-	◆	-	-

Total number of glyphs shown from JetBrainsMono-Regular.otf: 1319

Literatur

[1] Wolfgang Beinert: Variable Fonts, 2021, www.typolexikon.de (besucht am 5. 7. 2024).

[2] Rolf Niepraschk: »Die Schriftfamilie ›JetBrains Mono‹«, *DTK*, 32.1 (2020), 37.

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: 15. 7. 2024)

Mathe-Handbuch

Seit dem 14. Juni ist das lang erwartete Handbuch »Mathematics in ConT_EXt« (Arbeitstitel »ConT_EXt for the working mathematician«) Teil der LMTX-Distribution. Der Autor Mikael Sundqvist lehrt Mathematik an der Universität Lund (Schweden) und hat ungefähr drei Jahre lang zusammen mit Hans Hagen den Mathematiksat in ConT_EXt gründlich überarbeitet.

Die A4-Version des Handbuchs hat 246 Seiten, die Bildschirm-Version über 470. Es enthält viele Beispiele von den Grundlagen über die zahlreichen unterstützten Mathe-Schriften bis zu Grafiken in MetaPost sowie eine ausführliche Liste von Symbolen (in der A4-Version allein 54 Seiten). Der trockene Stoff wird aufgelockert durch einige Karikaturen von Sofia Kockum mit den ConT_EXt-Maskottchen Kuh und Dodo. Während Hans Hagen das Werk als »vollständig« bezeichnete, meinte der Autor bescheiden, es enthielte »die wichtigsten Befehle«. Von den Anwendern gab es großes Lob.

Detailverbesserungen

Es gab Verbesserungen und Korrekturen in vielen Bereichen, die aber oft nur exotische Details betreffen.

- »meaningful math«¹: Seit dem BachoT_EX-Treffen können Formeln auch auf Ukrainisch ausgeschrieben werden, allerdings ist dies nicht so gut ausgearbeitet wie auf Englisch oder Schwedisch.
- »granular align«: Es gab Verbesserungen beim mehrstufigen Absatzumbruch.
- »stacking«: Ein neuer Mechanismus, sich wiederholende Grafiken aus mehreren Schichten zusammenzusetzen, die sich frei kombinieren lassen.
- »CSL«: Die Unterstützung für Zitationsstile ist in der Diskussion. Es gibt ein »proof of concept« von Max Chernoff auf Basis von »citeproc-lua«, das auch dem L^AT_EX-Paket »citation-style-language« zugrund liegt.

¹ Siehe »Aussprache von Formeln« in dieser Rubrik in der vorigen Ausgabe.

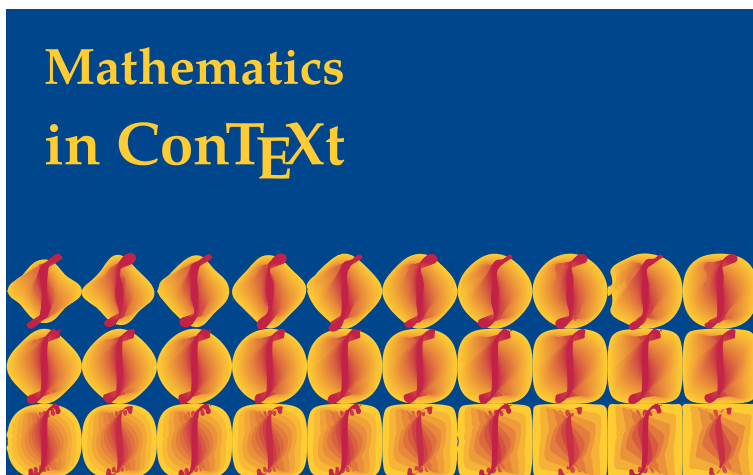


Abb. 1: Titelseite des Mathe-Handbuchs (Bildschirm-Version)

- Bei eigenen Umgebungen lässt sich der optionale Parameter für Einstellungen jetzt einfach aktivieren (siehe folgendes Beispiel von Wolfgang Schuster).

```
\definestartstop[Highlight][
  arguments=yes,
  style=mono,
  left>={\thinspace}]

\definedescription[description][
  width=broad,
  distance=\spaceamount,
  headcommand=\Highlight[right=:],
  alternative=hanging]

\starttext
This is a \Highlight{highlight}.

\startdescription[title={Lorem ipsum}]
\samplefile{lorem}
\stopdescription

\stoptext
```

Diskussion zur Barrierefreiheit

ConT_EXt kann Tagged PDF und damit PDF/A oder PDF/UA erzeugen. Allerdings wies nicht zuletzt Ulrike Fischer darauf hin, dass das erzeugte Tagging wenig brauchbar sei, auch wenn es sich formal validieren lässt. Das L^AT_EX-Team hat in letzter Zeit viel Mühe in die Erzeugung von besserem Tagged PDF investiert. Hans Hagen hat dagegen bisher keine Ambitionen, über ein formal korrektes Format hinauszugehen, da der Aufwand in keinem Verhältnis zum Nutzen stehe, die Spezifikation nicht eindeutig und die Vielfalt einer Makrosprache nie sinnvoll zu erfassen sei, von der Auswertung in PDF-Viewern ganz abgesehen. Statt pseudo-barrierefreie PDFs anzubieten, sei man mit HTML-Seiten besser beraten, die sich leicht aus dem XML-Export von ConT_EXt erzeugen lassen, wenn man nicht gleich von HTML oder einem medienneutralen Format ausgehen möchte.

Das Thema soll beim ConT_EXt Meeting im Zusammenhang mit PDF 2.0 und MathML weiter diskutiert werden.

ConT_EXt Meeting

Bei Erscheinen dieser Ausgabe ist das jährliche Treffen wahrscheinlich schon vorbei – diesmal haben wir es ans Ende der Ferienzeit gelegt, statt in den Herbst, um mal einem anderen Personenkreis die Teilnahme zu ermöglichen (manche können nur in den Ferien, andere nur außerhalb, ...). Der Wasserturm von Lutten bietet der zu erwartenden kleinen Gruppe einen geeigneten Rahmen, um sich über Entwicklungen und Projekte auszutauschen, den einen oder anderen »Hack« umzusetzen, alte Freundschaften zu pflegen und neue aufzubauen.

Aufgeschnappt

Luzia Dietsche

Immer mal wieder schnappe ich beim Surfen im Internet oder beim Lesen verschiedener Kommunikationslisten Hinweise auf, die vielleicht auch andere interessieren.

Herman Zapf Alphabete

Hermann Zapf dürfte Etlichen als genialer Schriftdesigner ein Begriff sein. Vor kurzem nun wurde eine Webseite freigegeben zu Ehren von Hermann und Gudrun Zapf-von Hesse:



This site takes a closer look at the life's work of the Typographers Hermann and Grudrun Zapf-von Hesse.

<https://zapf-alphabets.com/>

Otl Aicher

Otl Aicher war einer der wichtigsten deutschen Gestalter des 20. Jahrhunderts. Mit seinen Schriften, Plakaten, Logos und dem visuellen Konzept für die Olympischen Spiele 1972 in München prägte er das Erscheinungsbild der Bundesrepublik. Äußere Form und innere Haltung gehörten dabei immer zusammen.



Ihm ist ein Teil der Podcast-Reihe »BR2 Radiowissen« gewidmet: <https://www.br.de/mediathek/podcast/radiowissen/otl-aicher-gestaltung-mit-haltung/1855899>

Buch zu Web-Typographie

Jason Santamaria schreibt:

My book On Web Typography is now free!

It's 10 years old, and with A Book Apart winding down, I'd like to keep it available for everyone to read. Even though it's about ever-evolving web typography, the information is still solid – I wrote it to act more as a guide for approaching aesthetics and function, and less about the specifics of code. Enjoy!



<https://jasonsantamaria.com/#on-web-typography>

Von fremden Bühnen

Neue Pakete auf CTAN

Jürgen Fenn

Der Beitrag stellt neue Pakete auf CTAN seit der letzten Ausgabe bis zum Redaktionsschluss in umgekehrter chronologischer Reihenfolge vor. Bloße Updates können auf der moderierten *CTAN-ann*-Mailingliste oder als RSS-Feed auf <https://ctan.org/> verfolgt werden.

telprint und *hypcap* von der *Oberdiek Package Support Group* lagern die gleichnamigen Pakete aus dem *oberdiek*-Bundle aus. Dies steht im Zusammenhang mit kleinen Änderungen, um die Pakete mit dem $\LaTeX$ Tagged PDF Project kompatibel zu machen.

CTAN:macros/latex/contrib/telprint

CTAN:macros/latex/contrib/hypcap

pgfplotsthemebeamer von »Samcarter« ermöglicht es, die Farben des Beamer-Themes in *pgfplots* einzusetzen. Damit passen sich die Grafiken farblich nahtlos und harmonisch in die Präsentation ein. Die Farben können für eine einzelne Grafik oder für alle Grafiken in einem Dokument vorgegeben werden.

CTAN:graphics/pgf/contrib/pgfplotsthemebeamer

doibanner von *Christian Schreinemachers* erzeugt einen DOI-Banner mit Hyperlink zum Online-Volltext, der auch skaliert werden kann.

CTAN:macros/latex/contrib/doibanner

bonum-otf von *Herbert Voß* enthält die $\LaTeX$ -Unterstützung für die Schrift $\TeX$ Gyre Bonum.

CTAN:fonts/bonum-otf

wrapstuff-doc-en von *Siyu Wu* ist die chinesische Übersetzung der Dokumentation zu dem Paket *wrapstuff* von *Qing Lee*, die bis dahin nur auf Englisch vorlag. Das Paket steuert, wie Text eine Grafik umfließen kann.

CTAN:info/wrapstuff-doc-en

luwiantype von *Jonathan Walls* ermöglicht das einfache Eingeben und Setzen von Hieroglyphen-Luwisch mit der Schriftart Noto Sans Anatolian Hieroglyphic von Google.

CTAN:fonts/luwiantype

moremath von *Marcel Ilg* ist ein weiteres Paket, das zusätzliche Befehle für den Mathematiksatz zur Verfügung stellt.

CTAN:macros/latex/contrib/moremath

passopt von *Qu Yi* ändert bei der Weitergabe von Optionen an Klassen und Pakete die Reihenfolge, in der sie später ausgeführt werden.

CTAN:macros/latex/contrib/passopt

tango von *Michel Bovani* ist eine Dokumentenklasse für den Mathematikunterricht, die viele Pakete, die man oft braucht, automatisch lädt, und nützliche Umgebungen bereitstellt.

CTAN:macros/latex/contrib/tango

gelasiomath von *Michael Sharpe* erweitert den Font Gelasio um Zeichen für den Mathematiksatz und um kleine Kapitälchen auf der Grundlage von XCharter.

CTAN:fonts/gelasiomath

latexscreenshoter von *Quentin Vaney* ist eine Lösung, mit der man unter Linux Screenshots aus dem $\LaTeX$ -Dokument heraus zur Laufzeit erstellen und einbinden kann. Sowohl X als auch Wayland werden unterstützt.

CTAN:graphics/latexscreenshoter

tiet-question-paper von *Raghav B. Venkataramaiyer* ist eine Dokumentenklasse für Testfragebögen am Thapar Institute of Engineering and Technologie (TIET).

CTAN:macros/latex/contrib/tiet-question-paper

SpeLaTeX von *Walter Daems* ist Teil einer größeren Lösung, um Text, Formeln, Grafiken und Tabellen, die in einer mit $\LaTeX$ erstellten PDF-Datei enthalten sind, in gesprochene Audio-Dateien umzuwandeln und mit ihrem Ursprungstext durch Hyperlinks zu verknüpfen. Zum Erzeugen der Audios gibt es das Perl-Modul `SpeL::Wizard` auf CPAN. Per `babel` wird die richtige Aussprache gewählt (bisher stehen Deutsch und Englisch bereit). Das funktioniert mit den PDF-Viewern Okular, Acrobat Reader und PDF-Xchange sowie mit den Audio-Playern SoX, Totem und VLC Media Player. Ein Emacs Org-mode-Exporter ist angekündigt.

CTAN:macros/latex/contrib/spelatex

linearregression von *Battista Benciolini* ist ein weiteres Paket zur Regressionsanalyse, das die Ergebnisse sowohl als Tabelle als auch grafisch ausgeben kann.

CTAN:macros/latex/contrib/linearregression

standardsectioning von *Markus Kohm* stellt die üblichen Makros für Überschriften aus den Standardklassen bereit. Dadurch kann man Pakete verwenden, die dieser bedürfen und die deshalb sonst nicht mit KOMA-Script funktionieren.

CTAN:macros/latex/contrib/standardsectioning

xint-regression von *Cédric Pierquet* dient zur Regressionsanalyse. Die Berechnungen werden von dem Paket *xint* erledigt.

CTAN:macros/latex/contrib/xint-regression

ensps-colorscheme von *Aliaume Lopez* stellt die Farbpalette der École Normale Supérieure Paris-Saclay bereit, so dass man es in Abschlussarbeiten oder in Präsentationen verwenden kann.

CTAN:macros/latex/contrib/ensps-colorscheme

asy-overview von *Jim Hefferon* bietet eine knappe Einführung in das Zeichnen von mathematischen Grafiken mit Asymptote.

CTAN:info/asy-overview

rub-kunstgeschichte von *Joran Schneyer* ist eine Dokumentenklasse für Abschlussarbeiten am Kunstgeschichtlichen Institut der Ruhr-Universität Bochum.

CTAN:macros/latex/contrib/rub-kunstgeschichte

nxuthesis von *Xianyuan Bao* enthält eine Dokumentenklasse und einen BibTeX-Stil für Abschlussarbeiten an der chinesischen Ningxia-Universität.

CTAN:macros/unicodetex/latex/nxuthesis

mpkiviat von *Maxime Chupin* zeichnet Kiviat-Diagramme für statistische Auswertungen mit Hilfe von METAPOST.

CTAN:graphics/metapost/contrib/macros/mpkiviat

catppuccinpalette von *Lukas Heindl* stellt die vier *Catppuccin*-Pastellfarbpaletten von <https://catppuccin.com/palette> für die Verwendung mit dem Paket *xcolor* bereit.

CTAN:macros/latex/contrib/catppuccinpalette

semesterplannerlua von *Lukas Heindl* dient zum Zeichnen von Stundenplänen, Kalendern und Terminlisten zur Planung des Semesters, des Schuljahrs oder ähnlicher Projekte.

CTAN:macros/luatex/latex/semesterplannerlua

skillicons von *Anton Mnatsakanov* stellt Icons zur Verfügung, die man zum Beispiel in Bewerbungen einfügen kann, um seine Kenntnisse in Bezug auf Programmiersprachen, Plattformen oder Anwendungen zu veranschaulichen.

CTAN:macros/latex/contrib/skillicons

rigidnotation von *Philippe Nadeau* dient zum Setzen von Vektoren und Matrizen in der neuen sogenannten RIGID-Notation nach ISO 80000, die 2023 eingeführt wurde.

CTAN:macros/latex/contrib/rigidnotation

colorblind von *Simon Pfahler* stellt zwei Farbpaletten (*Paul Tol* und *Okabe-Ito*) bereit, mit denen man barrierefreie Dokumente gestalten kann, die auch für farbblinde Menschen gut zu lesen sind.

CTAN:macros/latex/contrib/colorblind

letesansmath von *Daniel Flipo* enthält den OpenType-Mathematik-Font namens *LeteSansMath.otf* inklusive der dazugehörigen Unterstützung für Lua²_{TEX} und X₂²_{TEX}, den man zusammen mit dem TrueType-Font *Lato sans-serif* einsetzen

kann.

CTAN:fonts/lete-sans-math

tkz-grapheur von *Cédric Pierquet* stellt Makros bereit, mit denen man schöne Funktionsgraphen für den Mathematikunterricht in der Sekundarstufe mit Hilfe von *TikZ* zeichnen kann.

CTAN:graphics/pgf/contrib/tkz-grapheur

ximera von *Jim Fowler*, *Jeremiah Hocutt*, *Oscar Levin*, *Jason Nowell*, *Wim Obbels*, *Hans Parshall* und *Bart Snapp* ist eine Dokumentenklasse, die für die gleichnamige Lernplattform <https://ximera.osu.edu/> entwickelt wurde. Damit soll man Texte, die mathematische Formeln enthalten, möglichst leicht in PDF und zugleich in HTML veröffentlichen können (Arbeitsblätter, Handouts oder Übungen).

CTAN:macros/latex/contrib/ximera

ifis-macros von *Udo Wermuth* stellt zwei Befehle bereit, die in einem Aufsatz in TUGBoat 45:1 (2024) beschrieben wurden. `\ifisint` und `\ifisdim` testen, ob ein Eingabestring ein valider Integer oder eine valide Dimension für $\TeX$ sind.

CTAN:macros/plain/contrib/ifis-macros

Spielplan

17. 8. – 23. 8. 2024 **18th ConT_EXt Meeting**
Lutten, Niederlande
<https://meeting.contextgarden.net/2024/>



7. 9. 2024 **BayT_EX 2024**
Temporärhaus
Augsburger Straße 23–25
89231 Neu-Ulm
<https://baytex.in-ulm.de/>



3. 4. – 5. 4. 2025 **DANTE 2025**
Hochschule Darmstadt

Stammtische



In verschiedenen Städten im Einzugsbereich von DANTE e.V. finden regelmäßig Treffen von TeXies statt, die für alle 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.lists.rwth-aachen.de>

»Anvers«, Kockerellstr. 20, 52062 Aachen

Erster Donnerstag im Monat, 20:00 Uhr

Bad Doberan

Carsten Vogel

texnicher@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. 131 B, 10115 Berlin

Zweiter Dienstag im Monat, 19:00 Uhr

Darmstadt

Karlheinz Geyer

geyerk@posteo.de

»Poseidon«, Rheinstr. 41, 64293 Darmstadt

Erster Freitag im Monat, 19:00 Uhr

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

zur Zeit 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>

wechselnder Ort, i. d. R. letzter Freitag im Monat



Köln

Uwe Ziegenhagen

uwe@dante.de, *zur Zeit inaktiv, Interessenten bitte per Mail melden*

Leipzig

Erhard Pross

Erhard.Pross@gmx.de

nächstes Treffen am 24. Oktober 2024 18:00 Uhr

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

München

Leah Neukirchen

leah@vuxu.org

<https://stammtisch-muenchen.dante.de/>

Erste Woche in geradzahligen Monaten, wechselnde Wochentage und Orte



Stuttgart

Bernd Raichle

bernd.raichle@gmx.de

»Trollinger« beim Feuersee, Rotebühlstr. 50, 70178 Stuttgart

Zweiter Dienstag im Monat, 19:30 Uhr

Adressen

DANTE, Deutschsprachige Anwendervereinigung T_EX e.V.

Bergheimer Straße 147

69115 Heidelberg

Neue Adresse!

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	
	Oliver Rath	

Ehrenmitglieder

Peter Sandner	22.03.1990	Klaus Thull († 2012)	22.03.1990
Yannis Haralambous	05.09.1991	Barbara Beeton	27.02.1997
Luzia Dietsche	27.02.1997	Donald E. Knuth	27.02.1997
Eberhard Mattes	27.02.1997	Hermann Zapf († 2015)	19.02.1999
Joachim Lammarsch	12.04.2014	Rainer Schöpf	12.04.2014

Webserver und Mailingliste

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

FAQ

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

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

T_EXnische Fragen

beraterkreis@dante.de

ak-schule@dante.de

Autoren/Organisatoren

Doris Behrendt siehe Seite 70	[5]	Henning Hraban Ramm hraban@fieee.net	[59]
Luzia Dietsche 71394 Kernen dtkred@dante.de	[3,61]	Martin Sievers siehe Seite 70	[4]
Jürgen Fenn Neu-Isenburg juergen.fenn@gmx.de	[63]	Herbert Voß Herbert.Voss@FU-Berlin.de	[48]
Johan Larsson	[38]	Keno Wehr wehr@abgol.de	[10]
Leah Neukirchen leah@vuxu.org	[8]	Uwe Ziegenhagen Köln	[32]

Die T_EXnische Komödie

36. Jahrgang Heft 3/2024 August 2024

Impressum

Editorial

Hinter der Bühne

- 4 Grußwort
- 5 Bericht der Schatzmeisterin
- 8 Einladung zur BayT_EX 2024 in Neu-Ulm

Bretter, die die Welt bedeuten

- 10 L^AT_EX und Schulphysik 7: Optik
- 32 Diagramme mit TikZ: Ein Beispiel
- 39 Moloch: Die Wiederauferstehung des L^AT_EX-Beamer-Stils Metropolis
- 48 JetBrains Mono
- 59 ConT_EXt kurz notiert
- 61 Aufgeschnappt

Von fremden Bühnen

- 63 Neue Pakete auf CTAN

Spielplan

- 67 Termine
- 68 Stammtische

Adressen

- 71 Autoren/Organisatoren