

Die Pakete um Schuleze

Peter Breitfeld¹

Störck-Gymnasium
Bad Saulgau

17. Februar 2013

1. E-Mail: phbrf@t-online.de <http://www.pBreitfeld.de>

Vorbemerkung

Das Paket arbeitet mit den Standard- \LaTeX -Klassen `article`, `report` und `book` zusammen sowie mit den Klassen `beamer`, `prosper` und `revtex4-1` (siehe dazu Abschnitt [1.1.3.4 auf Seite 13](#)).

Das Paket unterstützt die beiden Schriftfamilien:

1. Computer Modern (Standard für \TeX , mittels `lmodern`)
2. Palatino mit Pazo-Fonts für Formelsatz. Diese Kombination hat echte Small-Caps und Minuskelziffern im Text-Modus. Dieses Dokument ist so gesetzt. Option `[palationOSF]`.

Der optische Randausgleich wird seit September 2006 mittels des Pakets `microtype` realisiert. Es ist flexibler und einfacher anzuwenden als das bisher benützte `pdfcrop`.

Seit Dezember 2008 ist auch die Unterstützung des Pakets `auto-pst-pdf` eingebaut, welches erlaubt, `PSTricks` und `psfrag` auch mittels `pdf \LaTeX` zu übersetzen. Beachte, dass `\pstricks` *nicht mehr* automatisch geladen wird.

Am \LaTeX Standardlayout wurde einiges verändert: Der Fettdruck in Überschriften, im Inhaltsverzeichnis und den anderen Verzeichnissen wurde entfernt; das Inhaltsverzeichnis hat keine rechtsbündigen Seitenzahlen mehr, was die Lesbarkeit erhöhen soll. Die Kolumnentitel erscheinen nun nicht mehr in Versalien, sondern in Groß-Kleinschreibung.

Wichtige Änderungen: Ab Dez. 2010 habe ich das Chaos der Vektormakros aufgeräumt. Manche endeten auf »`vec`«, »`Vec`«, »`vek`« oder »`Vek`«. Nun enden alle hier definierten Makros und Optionen auf »`vek`«.

Seit Feb. 2012 gibt es die Makros `\abk{zB}`, `\abk{dh}`, `\abk{ia}`, `\abk{ua}` nicht mehr. Sie wurden durch das allgemeinere Makro `\abk{...}` ersetzt. `\vH` existiert weiterhin. Ebenso sind die Befehle `\testsfb` und `\mathsbb` verschwunden zugunsten von `\textbb` und `\mathbb`.

Seit Mai 2012 wurden alle `\MZ` Befehle entfernt, weil ich sowieso immer Minuskelziffern verwende. Ebenso ist der Befehl `\mb` verschwunden, weil es ja schon \LaTeX `\mid` für dasselbe gibt. Alle diese Änderungen werden in meinem Emacs mittels des Menüpunkts »BRF/AUCTeX/Akb. umwandeln (schule2e)« ersetzt.

Es gibt nun auch `schule2eLua.sty`, was sich mit `Lua \LaTeX` übersetzen lässt. Änderungen werden wie hier gekennzeichnet und beziehen sich auf Übersetzung mit `schule2eLua`.

Inhaltsverzeichnis

- 1 *Das Paket schule2e* • 5
 - 1.1 Paketbeschreibung • 5
 - 1.1.1 Optionen • 5
 - 1.1.2 Eingebundene Pakete • 9
 - 1.1.3 Anpassungen • 9
 - 1.1.4 Biblio, Theorem, Index, Reftex, Hyperref • 13
 - 1.1.5 Neue Zähler • 16
 - 1.1.6 Neue Längen • 16
 - 1.1.7 Fußnotenverwaltung • 17
 - 1.1.8 Fraktur und Initiale • 18
 - 1.1.9 Aufgaben-Verwaltung • 19
 - 1.1.10 Strukturierungen • 20
 - 1.1.11 Schriftarten und -Größen • 22
 - 1.1.12 Sperrung von Text • 23
 - 1.1.13 Spezielle Zeichen und Kombinationen • 24
 - 1.1.14 Formeln und Einheiten • 25
 - 1.1.15 Sonstige Makros • 29
 - 1.1.16 Tabellenbearbeitung • 29
- 2 *Weitere Pakete* • 33
 - 2.1 Das Paket typeareaB • 33
 - 2.2 Die Klasse briefbrf.cls • 34
 - 2.3 Das Paket tensindbrf • 36
 - 2.4 Das Paket palateuler • 36
- 3 *Externe Hilfsprogramme* • 38
 - 3.1 Eigene Routinen • 38
 - 3.1.1 Booklet und DviDuplex • 38
 - 3.1.2 DvizPdf • 39
- 4 *Hinweise zu Paketen* • 40
 - 4.1 Webo-Fonts • 40
 - 4.2 auto-pst-pdf • 40
- A *schule2eLua* • 42
- B *Obsoletes Umwandeln* • 45
- Index* • 46

Abbildungsverzeichnis

- 1.1 Code der multirow Tabelle • [32](#)
- 2.1 Code für Serienbrief • [36](#)

Tabellenverzeichnis

- 1.1 Tabelle der Optionen von schuleze • [5](#)
- 1.2 Eingebundene Pakete • [10](#)
- 1.3 Übersicht der Reftex-Einträge • [17](#)
- 1.4 Vordefinierte Spaltenstile • [30](#)
- 1.5 Zum MRzell Befehl • [32](#)

- 2.1 Seitenmaße in mm • [34](#)
- 2.2 Papiermaße in mm, Zoll und PostScript-Punkten • [34](#)
- 2.3 Befehle und Felder der Klasse briefbrf.cls • [35](#)

- 4.1 Ornamente aus WeboMintsGD • [41](#)

- A.1 Optionen von schulezeLua • [43](#)
- A.2 Von schulezeLua geladene Pakete • [44](#)
- A.3 unicode-math Schriftarten • [44](#)

1. Das Paket schule2e

1.1. Paketbeschreibung

1.1.1. Optionen

FEHLERWARNUNG

Wegen meiner Binderand-Option tritt bei Angabe von nicht vorhandenen (oder falsch geschriebenen) Optionen der Fehler »missing number, treated as zero« auf.

HAUPTOPTIONEN

[**alle**] Dies ist die Default-Einstellung und muss nicht angegeben werden. Alle hier aufgeführten Makros und Umgebungen sind verfügbar.

[**brief**] Dasselbe wie [alle], aber typeareaB- und sectsty-Einbindung sowie sonstige Layout-Änderungen finden nicht statt. Der Befehl `\parindentOff` (vgl. Abschnitt [1.1.3.3 auf Seite 12](#)) zeigt dann ebenfalls keine Wirkung, da die Briefklasse sowieso durch Leerzeilen getrennte Absätze verwendet. Die Einstellungen des Makros `\Minisecfont` gelten aber auch hier!

babel wird ebenfalls nicht geladen!

TAB. 1.1: Tabelle der Optionen von schule2e

Optionen für schule2e und schule2eLua:

alle	brief	helv		
noSectsty	withChapter	tradToc	noGLNrDot	
fnTrenner	pNFlush	sectNA	Pfeilvek	Gothicvek
AutoPST	AutoOff	<BCOR-Länge>		

Optionen nur für schule2e:

utf8	latin1	ansi	mac
noEco	palatino	palatinoOSF	Avant
pxBBold	SkriptFont	WasyFont	WasyInt
Eucal	orgGreek	noLinespread	

1. Das Paket `schule2e`

`[utf8]`, `[latin1]`, `[ansi]`, `[mac]` Mit diesen Optionen wird für `inputenc` »utf8«, »latin1«, »ansinew« bzw. »applemac« verwendet. Default ist »utf8«.
Lua Entfällt. Lua verwendet immer unicode.

LAYOUT OPTIONEN

`[noSectsty]` bewirkt, dass das Paket `sectsty` keine Änderungen der Überschriften durchführt, also Abschnittsüberschriften incl. `description`, `\Titelzeile`, `\NA` und `\NL` in Roman anstatt meiner eigenen Einstellungen gesetzt werden. Es wird dann stets das übliche \TeX -Verhalten, also fett gesetzt mit Serifen benützt. Die Einstellungen von `\Minisecfont` gelten aber weiterhin. Ebenso wird immer noch der Punkt nach den Gliederungsnummern gesetzt. Er muss mit `[noGLNrDot]` unterdrückt werden. Ist diese Option *nicht* gesetzt, dann wird dennoch bei Kapitelüberschriften auch das Wort »Kapitel« nicht mehr ausgegeben, allerdings habe ich »Teil« bei `\part` gelassen. (vgl. Abschnitt 1.1.3.1 auf Seite 9 und zu `revtex4-1` Abschnitt 1.1.3.4 auf Seite 13)

`[withChapter]` bewirkt, dass das Wort »Kapitel« bzw. »Anhang« in der Überschrift wieder auftaucht. In die Kolumnentiteln wird es dadurch immer noch nicht geschrieben; das will ich auch auf keinen Fall haben, weil es reine Platzverschwendung ist.

`[tradToc]` schaltet die \LaTeX -übliche Darstellung von Inhalts-, Abbildungs- und Tabellenverzeichnis ein. Ohne diese Option bekommt man die in diesem Dokument benützte Darstellung. Sie ist (meines Erachtens) freundlicher.

Zur \LaTeX -Darstellung schreibt CHARLES HEDRICK:

“What in the world is accomplished by putting the page numbers over on the right, and making people follow all those dots? Unless you think someone is going to take the average of all page numbers, there’s no need to line them up in one column.”

`[noGLNrDot]` verhindert den abschließenden Punkt nach den Gliederungsnummern von `\chapter`, `\section`, ...

`[noLinespread]` Bei Verwendung von Palatino-Schriften wird immer der Durchschuss mittels des Befehls `\linespread{1.04}` leicht erhöht. Dadurch entsteht ein Durchschuss von 12.5, 14.1 bzw. 15 Punkten bei Schriftgrößen von 10, 11 und 12 Punkt resp. Diese Option schaltet dieses Verhalten ab. Mit Option `[brief]` findet diese Sperrung nie statt.

Lua `schule2eLua` verwendet die Schrift TeX Gyre Pagella als Grundschrift. Dort habe ich die Sperrung *fest* eingebaut. Falls man sie unbedingt ändern will, sollte man im Vorspann eben `\linespread{faktor}` passend anpassen.

`[fnTrenner]` Mit dieser Option wird eine Trennlinie zwischen Text und Fußnoten gesetzt. Default ist diese hier ausgeschaltet. Siehe auch Abschnitt 1.1.7 auf Seite 17.

[pNflush] Mit dem Befehl Kopfzeile werden default die Seitennummern zentriert in den Fuß gesetzt. Mit dieser Option werden sie außen in den Fuß gesetzt.

[sectNA] Aufgabennummern werden als Unternummern der Sectionnummer angegeben, also in der Form 3.7 für die 7. Aufgabe in Section 3.

[<Binderand-Länge>] Das ist ein übler Trick, um für typeareaB (vgl. Abschnitt 2.1 auf Seite 33) einen Binderand einzustellen. Wird irgendwo in der Liste der Optionen eine Maßeinheit (z. B. 12 mm ist oft nicht schlecht) angegeben, dann wird diese Länge als Binderand betrachtet (default ist der Binderand 0). Dies sollte höchstens einmal geschehen. Alle schule2e ansonsten unbekanntenen Optionen werden damit als solche Längeneinheiten betrachtet und versucht, der Länge brfBCOR zuzuweisen. Jede Nicht-Länge gibt dann einen Übersetzungsfehler (»Missing number, treated as zero«), auch ein Schreibfehler bei Setzen einer richtig gemeinten Option!

Setzt man diesen Binderand auf den unsinnigen **speziellen Wert 111 in** (ca. 3 m), so erhält man ein Layout ohne Binderand-Korrektur mit gleichen linken und rechten Rändern. Das ist für Dokumente gedacht, die ein twoside-Layout haben, aber üblicherweise nur auf dem Bildschirm gelesen werden. Insbesondere bleiben in diesem Fall die Header-Einstellungen: »auf geraden Seiten die ›höchste‹ Gliederungsebene, auf ungeraden Seiten die ›zweithöchste‹ Ebene«, erhalten. Dieses Dokument wurde so übersetzt.

AUSWAHL VON SCHRIFTEN UND SONDERZEICHEN

Hier sind *alle* Schriften vordefiniert: Hauptfont ist TeX Gyre Pagella (mit Minuskelziffern), Sans Serif ist Myriad Pro und Typewriter ist DejaVu Sans Mono. Mathematischer Zeichensatz stammt aus den Fonts TeX Gyre Pagella Math, XITS (für `\mathcal` und `\mathscr`) sowie Asana-Math (für `\mathbb`). Griechische Kleinbuchstaben sind aus Latin Modern Math. Lua

[noEco] schaltet die Verwendung des Pakets cfr-lm aus. cfr-lm verwendet lmodern mit (nichtproportionalen) Minuskelziffern (für `rm` und `sf`) Mit [noEco] werden normale lmodern Fonts benützt (ohne Minuskelziffern). Die Option wirkt sich nur aus, falls die Option [palatino(OSF)] nicht gesetzt ist.

Entfällt. Lua

[palatino], [palatinoOSF] lädt das eigene Paket palatuler (siehe Abschnitt 2.4 auf Seite 36) mit der Option [ppl] (ohne Minuskelziffern) bzw. der Option[pplj] (mit Minuskelziffern). Grundschrift ist Palatino (`rm`), Myriad (`sf`) und lmtt (`tt`).

Entfällt. Lua

[helv] bzw. [Avant] Diese Optionen werden an das Paket palatuler weitergereicht, und bewirken die Verwendung von Helvetica bzw. Avant Garde anstelle der dort default verwendeten Myriad als Serifenlose. *Muss* gesetzt werden, wenn man keine Myriad hat.

Nur [helv] existiert; nimmt TeX Gyre Heros statt Myriad. Lua

1. Das Paket `schule2e`

`[pxBbold]` Es sollen die `pxsyb`-Fonts für Blackboard-Zeichen ($\mathbb{A}\mathbb{B}\mathbb{C}\dots$) verwendet werden. Diese Option *kann* gewählt werden, wenn dieser Font installiert ist. *Nur* bei Computer Modern Fonts hat diese Option eine Wirkung.

Lua Entfällt.

`[SkriptFont]` lädt die Skript-Buchstaben aus dem Paket `mathrsfs`.

Lua Entfällt. Der Font ist eingebaut.

`[WasyFont]` lädt die Wasy-Fonts für diverse Sonderzeichen *ohne* die dort definierten Integrale.

`[WasyInt]` lädt die Wasy-Fonts *mit* den dort definierten Integralen.

Lua In Lua muss der `WasyFont` (`[nointegrals]{wasysym}`) separat geladen werden. Die Wasy-Integrale sollte man hier nicht verwenden. Siehe Anhang A auf Seite 42.

`[Eucal]` lädt die `eucal` Fonts (aus \mathcal{AMS}) für kalligraphische Zeichen mittels `mathcal`. Ist diese Option *gesetzt*, steht auch der Befehl `\CMcal` zur Verfügung, der auf »normale« calligraphische Buchstaben umschaltet.

Lua Entfällt.

`[Pfeilvek]`, `[Gothicvek]` Standardmäßig werden in `schule2e` Vektoren mittels `\vek italic fett` gesetzt, also so: a, \hat{a} . Ist `[Pfeilvek]` gesetzt, dann werden Pfeile benützt (\vec{a}, \hat{a}). Ist `[Gothicvek]` gesetzt, werden Vektoren aus lateinischen Buchstaben in Fraktur geschrieben, während z. B. Vektoren mit griechischen Buchstaben einen Pfeil bekommen ($\vec{\omega}, \hat{\omega}, \mathbf{v}, \hat{\mathbf{v}}$).

`[orgGreek]` Aus Schönheitsgründen werden die folgenden griechischen Buchstaben bei Voreinstellung `[palatino(OSF)]` aus einer anderen Schriftart genommen: `\varpi = \varpi` aus `cm`, `\varrho = \varrho` und `\xi = \xi` aus `pxr`. Eine fette Variante dieser Buchstaben bekommt man *nur* als `\bvarpi = \bpi v = \varpi`, `\bvarrho = \brhov = \varrho`, `\bxi = \xi`. Bei `lmodern` werden deren Zeichen genommen (auch die fetten, wegen Kompatibilität). Setzung dieser Option schaltet die Umdefinierung ab.

Lua Entfällt. Alle kleinen griechischen Buchstaben werden aus `Latin Modern Math` genommen. Siehe Anhang A auf Seite 42.

WEITERE OPTIONEN

`[AutoPST]` lädt das Paket `auto-pst-pdf` (siehe Abschnitt 4.2 auf Seite 40). Falls auf dem \LaTeX -Weg (`tex`→`dvi`→`ps`→`pdf`) übersetzt wird, hat das Paket keine Wirkung, es stellt dann nur ein paar Makros wie `\mathfig` zur Verfügung. Dieses Paket wird *nur* benötigt, wenn man `\pstricks` oder `\psfrag` verwendet. Umwandlung von `eps` in `pdf` macht heutiges `pdf\text{\LaTeX}` automatisch.

`[AutoOff]` übergibt den Parameter »off« an das Paket `auto-pst-pdf`. Dadurch wird verhindert, dass bei einer Übersetzung mit `pdf\text{\LaTeX}` (`tex`→`pdf`) eine Umwandlung von Bildern in `pdf` erfolgt. Diese Option kann man wählen, wenn keine *weiteren* Bilder, `pstricks`-Dinge usw. mehr zu bearbeiten sind; so kann die Übersetzungszeit verringert werden. Siehe auch Abschnitt 4.2 auf Seite 40.

[ora] Mit dieser Option wurde *optischer Randausgleich* eingestellt. Dazu wird das Paket *microtype* geladen. Das Dokument *muss* dazu mit *pdf_latex* übersetzt werden, was heutige T_EX-Distributionen immer machen. Dieses Dokument ist so gesetzt. Das ist seit 2.5.2012 *default*, ich habe die Option nur aus Kompatibilitätsgründen gelassen.

1.1.2. Eingebundene Pakete

Da hier viele Pakete eingebunden werden, die zum Teil in ganz bestimmter Reihenfolge zu laden sind (z. B. *babel* nach *bm* und die Fontauswahl vor *tabellen*), wurde nahezu alles Wichtige gleich in *schule2e* aufgerufen. Der Aufruf erfolgt in der Reihenfolge von Tabelle 1.2 auf der nächsten Seite.

marvosym: Der Stil stellt eine Zahl weiterer Zeichen zur Verfügung.

wasysym: (nur mit Option [WasyFont] bzw. [WasyInt]). Stellt eine Reihe weiterer Sonderzeichen zur Verfügung. Inzwischen liegen die Wasy-Fonts als PostScript vor, so dass sie auch in PDF-Dateien gut verwendet werden können.

Muss händisch geladen werden. Siehe Anhang A auf Seite 42.

Lua

[english,ngerman] *babel*: (nur ohne Option [brief]). Deutsche Sprachanpassungen (neue deutsche Rechtschreibung).

[T1,obeyspaces,spaces] *url*: Eingabe von Web-Adressen, ohne Verrenkungen machen zu müssen. Ausgegeben werden sie in *\ttfamily*-Schrift. *obeyspaces* bewirkt, dass Leerzeichen nicht »geschluckt« werden und *spaces*, dass nach Leerzeichen umbrochen werden darf. *Vorsicht*: Nach einem Zeilenumbruch innerhalb von *\url* bzw. *\URL* im Quelltext muss man dann ganz vorne anfangen oder die Vorzeile mit einem Kommentarzeichen (%) beenden, sonst werden *alle* Leerzeichen mit in die Ausgabe übernommen! Beachte auch die Ausführungen zu *hyperref* im Abschnitt 1.1.4 auf Seite 15.

Manche hier vorhandenen Befehlsnamen gibt es schon im Paket »*circ*«. Es sind dies *\U*, *\I*, *\D* und *\G*. Innerhalb einer *circuit*-Umgebung sind dann stattdessen die Befehle *\sU*, *\sI*, *\sD* und *\sG* zu verwenden.

Das Paket *circ* wird nicht mehr unterstützt. Man nehme stattdessen *pst-circ*.

Lua

1.1.3. Anpassungen

1.1.3.1. Layout-Änderungen

Die Überschrift der *abstract*-Umgebung wurde von »Zusammenfassung« auf leer undefiniert. Man erkennt auch ohne Überschrift, dass hier ein Abstract steht. Umsetzung:

```
\addto\captionsgerman{\renewcommand{\abstractname}{irgendwas}}
```

1. Das Paket *schule2e*

Tab. 1.2: Von *schule2e* eingebundene Pakete. Pakete mit * werden von *schule2eLua* *nicht* eingebunden. (+/−) ist die Option zur Aktivierung/Deaktivierung, » → « zeigt, was automatisch mit geladen wird.

Paket	mit Option	Zweck	+/- Option
<i>inputenc*</i>	<i>utf8</i>	Umlaute direkt eingeben	
<i>fontenc*</i>	<i>T1</i>	Pflicht	
<i>xspace</i>	--	automatischer Space	
<i>units</i>	--	Einheiten und Brüche → <i>ifthen</i>	
<i>fancybox</i>	--	für Umgebung merken	
<i>ifpdf</i>	--	intern	
<i>subscript</i>	--	<i>textsubscript-</i> Makro	
<i>sectsty</i>	--	Stil von Überschriften	(−) <i>noSectsty</i>
<i>marvosym</i>	--	Sonderzeichen	
<i>mathtools</i>	--	→ <i>amsmath</i>	
<i>amssymb*</i>	--	math Sonderzeichen	
<i>amsthm</i>	--	Satz, Definition usw.	
<i>eucal*</i>	<i>mathcal</i>	»cal« Buchstaben	(+) <i>Eucal</i>
<i>braket</i>	--	Dirac-Klammern	
<i>textcomp*</i>	--	TS ₁ Zugriff	
<i>mathrsfs*</i>	--	Skript Buchstaben	(+) <i>SkriptFont</i>
<i>wasysym*</i>	s. u.	Wasy Paket	(+) <i>WasyFont/Int</i>
<i>palateuler*</i>	<i>ppl(j)</i>	<i>mathpazo</i> verwenden	(+) <i>palatino(OSF)</i>
<i>exscale*</i>	--	nur bei <i>lmodern</i> Fonts	
<i>lmodern*</i>	--	»neue« cm-Fonts	
<i>cfr-lm*</i>	--	OSF Ziffern bei <i>lmodern</i>	(−) <i>noEco</i> → <i>lmodern</i>
<i>typeareaB</i>	--	Seitenlayout	
<i>xkeyval</i>	--	»rechtzeitig« laden	
<i>array</i>	--	Tuning von Tabellen	
<i>rotating</i>	--	→ <i>graphicx-bundle</i>	
<i>numprint</i>	<i>np, boldmath</i>	»lined« Tabellenzahlen	
<i>tabularx</i>	--	automatische Weiten	
<i>booktabs</i>	--	hübsche Tabellen	
<i>multirow</i>	--	Spalten überspannen	
<i>bm*</i>	--	bold Math	
<i>auto-pst-pdf</i>	--	für <i>pstricks</i> und <i>psfrag</i>	(+) <i>AutoPST</i>
<i>babel</i>	s. u.	Sprachanpassungen	(−) <i>brief</i>
<i>url</i>	s. u.	Internet-Adressen	
<i>parskip</i>	--	Paragraph-Einrückung	(+) <i>\parindentOff</i>
<i>microtype</i>	--	Sperrung, Randausgleich ₁₀	
<i>icomma</i>	--	Space nach Komma	
<i>etoolbox</i>	--	intern	

In der zweiten Ebene der `enumerate`-Umgebung wurden die Nummerierungen von (a), (b),... auf a), b),... umgestellt.

Bei der `itemize`-Umgebung werden folgende Markierungen für die vier Ebenen eingestellt: Eins (●), Zwei (▷), Drei (★), Vier (◦). Typographisch sind mehr als zwei Ebenen zweifelhaft!

Das Aussehen von Inhalts-, Abbildungs- und Tabellenverzeichnis wurde verändert. Das \LaTeX -original-Verhalten kann mit der Option `[tradToc]` wiederhergestellt werden. Es sieht nun so aus wie in diesem Dokument.

Sowohl in den Kapitelüberschriften von `\chapter` als auch in den zugehörigen Kolumnentiteln wurde das Wort »Kapitel« entfernt. Dies kann nur durch völligen Verzicht auf die neuen Anpassungen mittels der Option `[noSectsty]` umgangen werden. In den Kolumnentiteln bekommt man das Wort »Kapitel« allerdings auch dadurch nicht mehr hin! Die Form von `\part` habe ich unverändert gelassen. Manche wollen unbedingt das Wort »Kapitel« bzw. »Anhang« (oder deren sprachliche Äquivalente) in den Überschriften (nicht den Kolumnentiteln) haben. Dazu dient die Option `[withChapter]`.

Die Kolumnentitel werden von \LaTeX normalerweise für die höchste Gliederungsebene in Versalien gesetzt. Das geschieht nun nicht mehr; es wird durchgängig Groß-Klein-Schreibung benützt.

Gleitumgebungen: Für die Gleitumgebungen sind in `schule2e` folgende Anpassungen durchgeführt:

```
\setcounter{topnumber}{3}
\setcounter{bottomnumber}{1}
\setcounter{totalnumber}{5}
\renewcommand*{\topfraction}{0.8}
\renewcommand*{\bottomfraction}{0.5}
\renewcommand*{\textfraction}{0.15}
\renewcommand*{\floatpagefraction}{0.6}
\renewcommand*{\dbltopfraction}{0.8}
\renewcommand*{\dblfloatpagefraction}{0.6}
```

1.1.3.2. Font-Einstellungen

Fette Überschriften wurden praktisch völlig entfernt. Dadurch sieht das Dokument meines Erachtens edler aus. Ein anderer Font und verschiedene Größen reichen zur Hervorhebung der Überschriften. Dazu wird das Paket `sectsty` benützt. Auch der – von mir gewünschte – Punkt nach Gliederungsnummern wird gesetzt. Beachte dazu auch die Optionen `[noSectsty]` und `[noGLNrDot]`.

Zum Einstellen der Fonts wurden folgende Makros definiert, die mittels `\renewcommand*` beliebig umgestellt werden können. Ihre Voreinstellungen sind:

```
\newcommand*{\Abschittfont}{\sffamily\mdseries}
```

1. Das Paket *schule2e*

```
\newcommand*{\subsubsectionfont}{\Abschnittfont\itshape}
\newcommand*{\Paragraphfont}{\mdseries\itshape}
\newcommand*{\Labelingfont}{\sffamily\mdseries}
\newcommand*{\Kopfzeilenfont}{\small\slhape\vphantom{fgq}}
\newcommand*{\Captionfont}{\small\sffamily}
\newcommand*{\textMinisecfont}[1]{\mccaps{#1}}
\newcommand*{\textAufgabenfont}[1]{\caps{#1}}
\newcommand*{\mathCaptionfont}[1]{\ensuremath{\mathsf{#1}}}
\newcommand*{\textCapTitFont}[1]{\{\rmfamily\textAufgabenfont{#1}}}
```

Das Makro `\Abschnittfont` legt dabei den Font für die Gliederungsebenen `\part` bis `\subsection` fest, sowie den Font für die Punkte der `description`-Umgebung und für das Makro `\Titelzeile`.

Da der Fettdruck weggefallen ist, musste der Font für `subsubsection`, der üblicherweise normale Größe hat, hervorgehoben werden. Dies geschieht durch kursive Schrift, die bei serifenlosen eher einem `slanted` entspricht.

Das Makro `\Paragraphfont` bestimmt das Aussehen von `\paragraph` und `\subparagraph`.

Das Makro `\Labelingfont` legt den Font für die Items der `labeling`-Umgebung (siehe Abschnitt [1.1.10 auf Seite 20](#)) fest.

Das Makro `\Kopfzeilenfont` bestimmt den Font der »Headings«. Das `\vphantom` dient dazu, den Trennstrich stets auf gleicher Höhe zu halten.

Die Makros `\textMinisecfont` bzw. `\textAufgabenfont` legen die Stile für `\minisec` bzw. `\NA` und `\NL` fest. Benützt man je die Option `[palatino]`, die ja nur gefakte Kapitälchen hat, sollte man `\textMinisecfont` und `\textAufgabenfont` passend umdefinieren. **Die verwendete TeX Gyre Pagella hat echte Kapitälchen.**

Lua

`\Captionfont` bestimmt den Font für captions – wer hätte das gedacht.

`\mathCaptionfont` wird vom Einheiten-Befehl `\U` (vgl. [1.1.14 auf Seite 27](#)) benützt, um die Einheiten in Captions im gleichen Font wie den Caption-Text auszugeben.

`\textCapTitFont` ist für die `caption`-Überschriften »Abb.« und »Tab.«.

Lua Folgendes wurde abweichend definiert:

```
\newcommand*{\Kopfzeilenfont}{\sffamily\small\itshape\vphantom{fgq}}
\newcommand*{\mathCaptionfont}[1]{\ensuremath{\mathsfup{#1}}}
```

1.1.3.3. Sonstige Anpassungen

Der Schalter `\parindentOff` lädt das Paket `parskip`, welches die Einrückung am Absatzbeginn ausschaltet und dafür die Absätze durch vertikalen Leerraum trennt. In älteren Versionen hatte ich das »von Hand« gemacht, dadurch wurde aber vertikaler Leerraum auch dort eingefügt, wo man ihn nicht will, etwa im Inhaltsverzeichnis und in Listen.

Der Zwischenraum hinter dem Dezimalkomma im mathematischen Modus wurde entfernt, ein Leerzeichen hinter dem Komma bringt ihn wieder zum Vorschein. Man

vergleiche $f(x, y) = f(x, y)$ mit $f(x, y) = f(x, y)$. Andernfalls würden Kommazahlen mit Zwischenraum nach dem Komma ausgegeben (10,34), was bescheiden aussieht. Hierzu wird `icomma.sty` geladen.

In allen unterstützten Schriftfamilien erscheinen nun große griechische Buchstaben kursiv (wie es sich gehört). Die für Einheiten benötigten aufrechten Ω und μ erreicht man über die Befehle `\Ohm` und `\mur`. Braucht man **aufrechte griechische Großbuchstaben**, so stelle man ein »up« voran, also `\upGamma` usw.

In Lua kann man – zumindest auf Mac – auf griechische Tastatureingabe umschalten. Lua
Dann kann man griechischen Text normal eintippen.

Fußnoten werden nun mit einer hängenden, nicht hochgestellten Fußnotenziffer gesetzt. Weitere Absätze in einer Fußnote bekommen einen Einzug. (vgl. Abschnitt [1.1.7 auf Seite 17](#))

1.1.3.4. Klasse `revtex4-1`

Die Klasse `revtex4-1` ist eine hübsche Klasse in zweispaltigem Satz. Meine Versuche mit dieser Klasse zusammen mit `schule2e` haben ergeben, dass das Gerüst etwa so aussehen sollte:

```
\documentclass{11pt,a4paper,twocolumn,nofootinbib}{revtex4-1}
\usepackage[optionen]{schule2e}
<ggf. weitere Pakete>
\usepackage{hyperref}
\bibliographystyle{<bst-file>} % muss vor \begin{document}
\begin{document}
<title, abstract>
\maketitle
<Material>
\bibliography{<bib-file>}
\end{document}
```

Die von `sectsy` gemachten Einstellungen werden bei dieser Klasse nicht durchgeführt, ebenso wird `typeareaB` nicht eingebunden. Auch wird `url` nicht zusätzlich vor `hyperref` geladen. Die Klasse hat kein `\chapter`. Fußnoten werden in Bib_TE_X integriert.

1.1.4. Biblio, Theorem, Index, Reftex, Hyperref

ANMERKUNG ZUR BIBLIOGRAPHIE

Für die Bibliographie verwende ich den `\bibliographystyle{dtk}`, das ist ein numerischer Stil, der fast dem originalen L^AT_EX entspricht, nur mit leichten Anpassungen für deutsch. Er muss ggf. von DANTE (Paket `dtk`) heruntergeladen werden. Ich habe ihn minimal zu `brf-dtk` abgewandelt, so dass er die Autoren in Kapitälchen setzt.

ANMERKUNG ZU THEOREM

Mit dem Paket `amsthm` kann das Aussehen von Satz-ähnlichen Strukturen angepasst werden. Das Paket erweitert die \LaTeX -`\newtheorem` Umgebung. Im Vorspann definiert werden können sie in den folgenden Varianten:

- (1) `\newtheorem{satz}{Satz}`
- (2) `\newtheorem{satz}{Satz}[section]`
- (3) `\newtheorem{definition}[satz]{Definition}`

Hier richtet (1) eine Umgebung `satz` mit dem Titel »Satz« ein. Die Nummerierung läuft fortlaufend ab 1 durch das ganze Dokument.

Dagegen wird in (2) in der Form »Satz 2.5« nummeriert, mit der »2« für die `section`. Im Falle (3) wird analog die Umgebung `definition` definiert. Die Option `[satz]` bewirkt, dass Sätze und Definitionen mit derselben Nummer durchgezählt werden. Definition und Aufruf erfolgen dann so:

```
\theoremstyle{brfplain}
\newtheorem{satz}{Satz}[section]
\theoremstyle{brfbsp}
\newtheorem{bsp}[satz]{Beispiel}
\begin{satz}[Weierstraßscher Produktsatz] bla bla \end{satz}
\begin{bsp}[Produktintegration] bla bla\end{Beispiel}
```

mit den Ausgaben:

Satz 1.1.1 (Weierstraßscher Produktsatz) *bla bla*

Beispiel 1.1.2 (Produktintegration) *bla bla*

Durch den Schalter `\theoremstyle{Stil}` kann – wie oben gezeigt – die Darstellung beeinflusst werden. Folgende Stile sind definiert:

- `plain`: Body »*italic*«, Head »**fett**« (default).
- `definition`: Body »roman«, Head »**fett**«.
- `remark`: Body »roman«, Head »*italic*«.
- `brfplain`: Body »*italic*«, Head »**sans-serif fett**«.
- `brfbsp`: Body »roman«, Head »**sans-serif fett**«.

Die letzten beiden sind von mir definiert, um ein konsistenteres Aussehen für mein Layout zu haben. Obwohl ich nahezu überall den Fettdruck aus dem Layout verbannt habe, lasse ich ihn hier leben, denn Sätze und Definitionen (für diese verwende ich `brfplain`) sowie die (mit `brfbsp` gesetzten) Beispiele sollen schnell gefunden werden können.

Der Schalter `\swapnumbers` schreibt die Nummer vor statt hinter die Überschrift.

In meinen `.emacs`-Einstellungen werden von `reflex` die folgenden Umgebungen erkannt: `Satz`, `Definition`, `Lemma` und `Beispiel`.

ANMERKUNG ZUR INDEXERSTELLUNG

In der Präambel muss der Befehl `\usepackage{makeidx}\makeindex` stehen. Dadurch wird beim \LaTeX Lauf das `.idx`-File erzeugt. Dieses muss durch das externe Programm `makeindex` gejagt werden. Ein anschließender \LaTeX -Lauf fügt dann den Index an der Stelle ein, wo das Kommando `\printindex` steht (also meist am Ende des Dokuments). Will man »Index« ins Inhaltsverzeichnis aufnehmen, so muss vor `\printindex` (ein Befehl aus `hyperref`) stehen:

```
\phantomsection\addcontentsline{toc}{section}{Index}
```

Um `\makeindex -c -g -s deutsch` (das Kommando `Index` in meiner AUC-TeX Konfiguration unter Emacs ruft diese Befehlssequenz auf) zur korrekten deutschen Anordnung ermöglichen zu können, wurde das File

```
$HOMETEXMF/makeindex/deutsch.ist
```

erstellt, in welchem das übliche Quote-Zeichen (") auf (>) umdefiniert wurde, damit Umlaute eingebbar werden. Dieses Quote-Zeichen nimmt den Sonderzeichen @, ! und | ihre Sonderbedeutung. Die Seitennummern werden dabei vom Stichwort mit einem Punkt (·) abgetrennt. Dieser `makeindex`-Aufruf kann getrost auch in Englisch verwendet werden.

Die Zeilen im File `deutsch.ist` sehen so aus:

```
quote '>'
delim_0 "$~\\Cdot ~$"
delim_1 "$~\\Cdot ~$"
delim_2 "$~\\Cdot ~$"
heading_flag 1
heading_prefix "\\textbf{"
heading_suffix "\\nopagebreak%\n"
symhead_positive "Symbole"
```

Zur Erinnerung:

- ! `\index{Pakete!Tabellen!tabularx}`
- @ `\index{so wird einsortiert@so steht's im Index}`
- | `\index{mathfonts|see{fonts}}`
erzeugt den Eintrag: »mathfonts · *siehe* fonts«
- `\index{blabla|()} auf Seite 12 und \index{blabla|)} auf Seite 19` erzeugen den Eintrag: »blabla · 12–19«
- `\index{hurrahurra|textit}` erzeugt den Eintrag: »hurrahurra · 19«
- > `\index{Klammeraffe (>@)}` erzeugt den Eintrag: »Klammeraffe (@) · 13«

ANMERKUNGEN ZU HYPERREF

`Hyperref` ist ein schönes Paket, das in `DVI`-, `PS`- und `PDF`-Dateien Referenzen zu Sprungmarken erzeugt. Dazu gehören auch die Einträge im Inhaltsverzeichnis, im Literaturverzeichnis oder im Index.

1. Das Paket `schule2e`

Dieses Paket lädt seinerseits auch das Paket `url`, aber zerstört dabei die wesentliche Eigenschaft des `\url`-Befehls, nämlich einen Zeilenumbruch zu ermöglichen. Erzeugt man direkt ein pdf-File mittels pdf \LaTeX , so werden die Links sauber umbrochen.

Um auch auf dem Weg `tex` \rightarrow `dvi` \rightarrow `dvips` \rightarrow `pdf` sauber umbrochene Links zu erzeugen, kann man in diesem Fall nach `hyperref` das Paket `breakurl` laden. Es hat einen neuen Befehl `\burl` und verwendet `\url` als Alias für diesen. Dieses »neue« `\url` bricht ggf. nach allen Sonderzeichenfolgen außer dem Bindestrich um. Die Verwendung dieses Pakets wird empfohlen. Um das Laden von `breakurl` nur dann stattfinden zu lassen, wenn keine Übersetzung nach PDF stattfindet, benütze ich in meiner Präambel:

```
\usepackage{hyperref}\ifpdf\else\usepackage{breakurl}\fi
```

Lua Man muss hier `\usepackage[unicode]{hyperref}` verwenden.

Von `schule2e` wird stets das Paket `url` mit den Optionen `[T1,obeyspaces,spaces]` geladen und der Befehl `\URL` als Alias des so implementierten `\url` definiert. Mit `\URL` wird immer sauber umbrochen, aber Links sind nicht anklickbar. Bei Verwendung der Klasse `revtex4` ist `\URL` das von `hyperref` definierte `\url`.

Will man mit `hyperref` Links zu Dateien erzeugen, kann man die Befehle `\href` oder `\url` verwenden. Das tut gut bei Links ins Internet. Bei Links auf lokale Dateien funktioniert anscheinend nur die `url`-Variante. Häufig ist aber der Pfad sehr lang oder man will ihn etwa bei Präsentationen nicht anzeigen lassen. Deshalb wurde der Befehl `\lokalhref` eingeführt:

```
lokalhref[faktor]{file:///path...}{Name}      Ausgabe: Name  $\rightarrow$  
```

In dem Kästchen steht eigentlich `\url{...}` aber dieser lange Link wurde mit `\scalebox` um `faktor` verkleinert. Default ist `faktor` 0.02, eine Veränderung ändert die Breite der Box. Das Kästchen kann man dann anklicken.

ANMERKUNGEN ZU REFTeX

Unter Emacs-AUCTeX werden durch Einträge in meiner `.emacs` die in Tabelle [1.3 auf der nächsten Seite](#) aufgeführten Umgebungen und Befehle von `reftex` erkannt.

1.1.5. Neue Zähler

`aufgabennr` ist der Zähler für Aufgaben und Lösungen für `\NA` und `\NL`.

`teilaufgabennr` ist der Zähler für Teilaufgaben. Er wird von den Umgebungen `TAufg`, `TTAufg` und dem Befehl `\TA` verwendet.

Daneben gibt es eine Reihe weiterer Zähler für den internen Gebrauch.

1.1.6. Neue Längen

`\breite`, `\Breite` sind Längen, die von den Befehlen `\Sumbreite` und `\Setzbreiten` verwendet werden.

`\TALength` ist die Boxbreite der a), b), ... von Teilaufgaben.

Tab. 1.3: Übersicht der RefTex-Einträge. Ein '#' in der Präfix-Spalte bedeutet, dass nach dem Präfix automatisch eine Nummer eingesetzt wird. Ansonsten wird eine editierbare Bezeichnung vorgeschlagen.

Befehl/Umgebung	Typ	Präfix	Schlüsselwörter
<i>eingebaute:</i>			
section-artige	s	sec:	part(s), chapter(s), chap., section(s) Teil(e), Kapitel, Kap., Abschnitt(e) line, Line, Zeile (für lineno.sty)
equation-artige	e	eq:#	equation(s), eqn., Gleichung(en), Gl.
figure-artige	f	fig:#	figure(s), fig(s), Abbildung(en), Abb.
table-artige	t	tab:#	table(s), tab., Tabelle(n), Tab.
Fußnoten	n	note:	footnote(s), note(s), Anmerkung(en), Anm.
\begin{enumerate}	i	item:#	item(s), Punkt(e)
<i>selbst definierte:</i>			
\figcaption	f	fig:#	wie figure-artige
\tabcaption	t	tab:#	wie table-artige
\begin{TAufg}	A	auf:#	Aufgabe, Aufg. Teilaufgabe, Teilaufg. Lösung, Lös., Lsg.
\begin{satz}	Z	satz:#	Satz, Theorem
\begin{def}	D	def:#	Definition, Def.
\begin{bsp}	B	bsp:#	Beispiel, Beisp., Bsp.
\begin{lem}	L	lem:#	Lemma, Lemmata, Lemmas
\TA, \NA und \NL	A	auf:#	wie bei \begin{TAufg}
\starlabel	e	eq:#	wie equation-artige

1.1.7. Fußnotenverwaltung

Das Layout der Fußnoten wurde verändert. Sie sehen nun so aus, wie in diesem Dokument. Auch gibt es default den Trennstrich nicht mehr. Mit der Option [fnTrenner] kann er aktiviert werden (für diejenigen, die ihn unbedingt wollen).

Die folgenden Befehle dienen der Aufgabe, in einem Text an mehreren Stellen eine Fußnotenziffer¹ anzubringen, die alle zur selben Fußnote¹ gehören. Das erste Auftreten einer solchen Fußnote wird mit \fn gesetzt, die folgenden Verweise mit fnmark. Unter Tabellen in Minipages² will man oft keinen Trennstrich² vor den Fußnoten¹, dieser ist default ausgeschaltet. In Minipages werden die Fußnotenziffern wie gewohnt in Buchstaben angezeigt. Da die zu referenzierenden Fußnoten auf einer früheren Seite liegen können, wurden sie anklickbar gemacht.

1. Das dient der Erklärung

2. auch außerhalb von Tabellen ist der Trennstrich laut Tschichold schlecht!

1. Das Paket `schule2e`

`\fn{refNr}{text}` erzeugt eine gewöhnliche Fußnote mit dem Text `text` und einen Label der Form `fn:refNr` (nicht anklickbar).

`\fnmark{refNr}` bringt eine Fußnotennummer am Text an, die auf dieselbe Fußnote verweist, wie das `\fn{refNr}{text}` (anklickbar).

1.1.8. Fraktur und Initiale

Verwendet wird `blackletter1`, eine Sammlung virtueller Fonts zum Zugriff auf die Fonts `yfrac`, `ygoth` und `ywab`, sowie auf die Initialen aus `yinit`. Im Unterschied zur originalen Kodierung haben diese Fonts eine T1-Kodierung, was die Eingabe und Trennmöglichkeiten sehr erleichtert. Dazu gibt es die folgenden Befehle und Umgebungen:


`\frakturOn` schaltet auf Schriftart Fraktur um und sollte nur innerhalb einer Umgebung eingefügt werden.

`\textfraktur{...}` schreibt Argument in Fraktur.

`\textInitial{...}` erzeugt einen Initialbuchstaben aus `yinit`.

`\begin{fraktur}...` alles in dieser Umgebung in Fraktur setzen. Diese Umgebung und der folgende Befehl sind wohl die wichtigsten Fraktur-Befehle.

`\Initial[f]` beginnt den Absatz mit einem Initial-Buchstaben. Ist der optionale Parameter `f` gesetzt, dann wird der Absatz in Fraktur geschrieben mit einem Fraktur-Initial, sonst in gewöhnlicher Schrift mit einem Initial in »Huge« und dem ersten Wort in »SMALLCAPS«

 Dieser Paragraph wurde mit `\Initial[f]` eingeleitet. Ich schreibe noch ein paar Zeilen dazu, damit man die Wirkung dieses schönen Befehls besser einschätzen kann. Wie man erkennt, werden vier Zeilen neben das Initial gestellt und dann unter dem Initial weitergeschrieben. Das Spacing wurde für Fraktur hingepflegt. Das oben eingefügte „Initial“ in lateinischer Schrift wurde mit dem Befehl `\textrm` eingefügt. Das sieht doch prächtig aus, oder nicht?

DIESER Absatz wurde mit dem Befehl `\Initial` begonnen, damit es schön aussehen soll. Um wirklich was zu erkennen, muss man schon ein paar Zeilen eingeben, sonst hängt das Initial unten raus, was ausgesprochen scheußlich aussehen würde.

Man kann die Größe des Initials anpassen, was manchmal günstig sein kann, wenn die Initiale Unterlängen haben. Dazu muss man ausführen:

```
\InitialScale{faktor}
```

Dabei wird das Initial um `faktor` (default: 1.3) vergrößert.

JEDER will ein schönes Initial als Beginn eines Absatzes sehen, muss aber beachten, dass der Paragraph nicht zu kurz ist. Dabei sollte man auch über die Größe des Initials nachdenken – es sollte nicht in den drunterstehenden Text hineinreichen.

Wir verkleinern das »J« etwas, indem wir den Faktor 1.1 wählen:

JEDER will ein schönes Initial als Beginn eines Absatzes sehen, muss aber beachten, dass der Paragraph nicht zu kurz ist. Dabei sollte man auch über die Größe des Initials nachdenken – es sollte nicht in den drunterstehenden Text hineinreichen.

Schriftvarianten innerhalb einer Fraktur-Umgebung ergeben die anderen Fonts:

`\emph` (oder `\itshape`) liefert gesperrte Fraktur. Obwohl Sperrung bei Antiqua böse ist, wird sie in Fraktur zur Hervorhebung verwendet.
`\textsl` (oder `\slshape`) liefert Schwabacher-Schrift.
`\textbf` (oder `\bfseries`) liefert Textur, also Gotische Schrift.
`\textrm` (oder `\rmfamily`) schaltet auf lateinische Zeichen um.
`\texttt` (oder `\ttfamily`) schaltet auf Typewriter-Schrift um.
`\textsf` (oder `\sffamily`) schaltet auf Sans-Serif um.

Das »ß« muss als "s eingegeben werden (andere Umlaute gehen). Die Schalter für nicht gebrochene Schriften benötigen ein vorangestelltes `\normalfont`. Lua

Gebrochene Schriften haben eine Vielzahl von Ligaturen. Normalerweise werden sie automatisch richtig eingesetzt. Sorgfalt ist allerdings bei »langem« und »rundem« Buchstaben »s« nötig. Ein rundes »s« wird automatisch am Wortende richtig eingesetzt, allerdings muss man es innerhalb eines Worts selbst einsetzen, so ist `Donners" | tag` für `Donnerstag` zu schreiben, also die Ligatur aufzuheben. Ein Doppel-s am Wortende existiert nicht. Dies wird entgegen der Neuen Deutschen Rechtschreibung stets durch ein »ß« ersetzt. Aber `Boss` kann als `Bos{s}` oder `Bos" | s` erzwungen werden.

Hier kann man auch ein langes Doppel-s am Ende erzeugen, indem man ein in diesem Font nicht vorhandenes Zeichen z. B. € ans Ende setzt. Lua

Anmerkung: Die Schriften `yfrak`, `ygoth` und `ywab` gibt es inzwischen auch im Type1-Format. Seit `texlive-2009` ist `blackletter1` mit allem Nötigen schon installiert.

1.1.9. Aufgaben-Verwaltung

`\mPunkte[lift]{Zahl}` schreibt als »marginpar« die Anzahl der Punkte in nebenstehender Form in den äußeren Rand. Der optionale Parameter `lift` ist default auf 0 pt. Er dient dazu, das Kästchen höher zu setzen. Damit kann man bei abgesetzten Gleichungen die Punkteangabe vertikal ausrichten. Dann muss `\mPunkte` nach der Gleichung bzw. Umgebung stehen, da dann noch ein `\vspace{-\belowdisplayskip}` angefügt wird. 7

`\Punkte{Zahl}` schreibt rechtsbündig z. B. (7P)

`\NA` bzw. `\NL` schreibt »AUFGABE« bzw. »LÖS. ZU AUFG.« und dahinter von 1 beginnend jeweils die Aufgabennummer, die automatisch erhöht wird. Die Nummern können mit `\NA{\label...}` referenziert werden. z. B. `\ref⇒1`.

Will man das Wort »Aufgabe« in einer eigenen Zeile stehen haben, dann sollte nicht `\NA\` verwendet werden, sondern mit einer Leerzeile oder `\par` ein neuer

1. Das Paket schule2e

Absatz begonnen werden, weil sonst am Seitenende nach dem Wort »Aufgabe« umbrochen werden könnte, was man ja nie will. Ein »\« würde das verwendete `\@afterheading` unterlaufen.

- `\TA` schreibt »a), b)...« für Teilaufgaben und zählt sie innerhalb einer Aufgabe oder Lösung automatisch hoch. `\TA` und `TAufg` können gemischt werden und zählen richtig weiter. Sie können mit `\TA{\label{...}}` referenziert werden. Das liefert z. B. `\ref` ⇒ 2b).
- `\TAspace` fügt den Zwischenraum ein, der von den Teilaufgaben a), b)... belegt wird.
- `\begin{TAufg}` beginnt eine Umgebung, deren Marken mit `\item` aufgerufen werden, was wieder Teilaufgaben hochzählt.
Referenziert wird mit `\item{\label{...}}`
- `\begin{TTAufg}[LS]{Spalten}` macht Aufgaben zeilenweise wie in einem Schulbuch. Intern wird eine `tabularx`-Umgebung benutzt, weshalb die Eingabe wie bei einer Tabelle erfolgt. `LS` erlaubt die Anfügung einer zusätzlichen letzten Spalte, etwa `[r]`, um Punkte anzugeben. Default ist `LS` leer. Die Tabelleneinträge sind automatisch im `Math-Modus`. Es gibt auch eine *Stern-Form*, bei der die Tabelleneinträge im `Text-Modus` sind.
- `\begin{TTAufgVar}{l_1}{l_2}{l_3}{l_4}` macht dasselbe wie `TTAufg`, hier kann allerdings die Länge der einzelnen, maximal vier Spalten angegeben werden, wobei nicht benötigte Spalten die Länge 0 bekommen müssen. Die Längen sind als Bruchteile der Länge einer `X`-Spalte anzugeben, ihre Summe muss gleich der Spaltenzahl sein; bei drei Spalten also z. B. `{0.5}{0.7}{1.8}{0}`. Wieder setzt die *Stern-Form* die Einträge im `Text-Modus`.
- `\iniAufg` setzt die Zähler von Aufgaben, Lösungen, Teilaufgaben zurück.

1.1.10. Strukturierungen

- `\clearempydoublepage` macht ein `\clearpage` und fügt ggf. eine leere Seite (ohne Kopf und Fuß) ein, so dass die folgende Seite rechts beginnt.
- `\Kopfzeile[x]{Außen}{Mitte}{Innen}` Dieser Befehl erzeugt eine Kopfzeile im Zeichensatz `\Kopfzeilenfont` (siehe Abschnitt [1.1.3.2 auf Seite 11](#)) mit oder ohne Trennstrich, mit bis zu drei Einträgen. Die Seitennummer steht immer im Fuß (Beachte auch Option `[pNflush]`). Der optionale Parameter (default ist `p`) kann sein:
 - H/h: Kopf wie beim `pagestyle{headings}`³ nur mit/ohne Trennstrich und Seitennummern im Fuß. Die Parameter Außen, Mitte und Innen werden bei dieser Option ignoriert.

3. Der `\pagestyle{headings}` musste dazu umdefiniert werden. Er schreibt nun immer die Kolumnentitel in Groß-/Kleinschreibung. Auch wurde das Wort »Kapitel« aus den Titeln verbannt.

K/k: Kopf enthält stets nur die höchste Gliederungsebene mit/ohne Trennstrich. Ansonsten wie H/h.

P/p: Ausgabe mit Seitennummern mit/ohne Trennstrich.

N/n: Ausgabe ohne Seitennummern mit/ohne Trennstrich.

`\begin{labeling}[Trennzeichen]{längster Titel}` erzeugt eine Listenumgebung, deren Ausgabe mit einem hängenden Einzug der Weite »längster Titel« erfolgt. (geklaut aus KOMA-Klassen). So wurde z. B. die obige Liste bei Kopfzeilen gesetzt.

`\begin{merkbbox}` ist eine `minipage` in einem Rahmen mit Schatten (unter Verwendung von `fancybox`) in Spaltenbreite. Es lassen sich nicht nur Textteile einschachteln, sondern auch Gleichungen, `eqnarray`-, `subequations` oder `align` Umgebungen usw. Dazu werden `\abovedisplayskip` und seine Verwandten für die Umgebung umdefiniert. Die Umgebung besitzt einen optionalen Parameter, der default den Wert `c` hat. Dadurch wird der Inhalt der `Minipage` horizontal im Rahmen zentriert. Will man dies nicht – etwa bei reinem Text – so wähle ein beliebigen anderen Buchstaben, etwa `l`. Rechtsbündig geht mit `r`.

`\minisec{txt}` Kleine Überschrift in `\Minisecfont`. (geklaut aus KOMA-Klassen). Beachte Option `[noSectsty]` und Abschnitt [1.1.3.1 auf Seite 9](#). Bei Option `[brief]` existiert dieser Befehl *nicht!*

`\begin{CBeschreibung}` erzeugt eine Liste, mit mehrzeiligen `\item`-Einträgen, die mittels `\URL` gesetzt werden. Die einzelnen Zeilen des Items werden durch zwei Kommata (,,) getrennt. Nach diesen sollte man ein Kommentarzeichen (%) einfügen, damit keine vorlaufenden Leerzeichen auftreten.

Für diese Umgebung wurde ein zwar übliches, aber vielen \LaTeX -Nutzern nicht geläufiges Konzept zur Verarbeitung von Text-Blöcken, die durch gewisse Trenner abgegrenzt sind, verwendet. Daher will ich eine Erklärung geben.

Die Item-Einträge sollen hier aus Blöcken bestehen, die durch »,,« getrennt sind. Ihre Anzahl ist zunächst unbekannt. Zuerst muss man eine »Ende-Marke« definieren:

```
\def\EndeMarke{dieEndeMarke}
```

Nun definiert man, wie die Zeile auszugeben ist

```
\long\def\Zeile#1{\xZeile#1,,dieEndeMarke,,}
```

An den Text, der in `#1` steht wird eine Trennmarke (,,) und die Ende-Marke angefügt und dann dem Makro `\xZeile` übergeben. Nun muss `\xZeile` definiert werden:

```
\long\def\xZeile#1,,{\def\temp{#1}
  \ifx\temp\EndeMarke    % jetzt sind wir durch
  \else \URL{#1}\par \expandafter\xZeile
  \fi}
```

1. Das Paket *schule2e*

In #1 steht nun der Textblock vor dem nächsten »,,«. Ist dies die Ende-Marke, so ist die Eingabe durchlaufen, also sind wir fertig. Im anderen Fall geben wir den Textblock mittels URL aus und beginnen eine neue Zeile. Das `\expandafter\xZeile` bewirkt, dass nun der Teil nach dem »,,« bearbeitet wird. Das `\def\temp` braucht man, damit man mit `\ifx` die Tokens vergleichen kann. Der Aufruf von `\Zeile` schreibt nun alle Blöcke untereinander.

`\raggedhyphen` schaltet Flattersatz mit Silbentrennung ein (linksbündig).

`\Titelzeile{txt}` Hier wird »txt« als eine zentrierte Überschrift ausgegeben. Für den Schriftstil wird `\Abschnittfont\Large` benützt. Beachte dazu auch den Abschnitt 1.1.3.1 auf Seite 9 und die Option [noSectsty]. Dieser Befehl besitzt den optionalen Parameter [v]. Wird dieser verwendet, so wird statt in Abschnittfont in Versalien ausgegeben (ohne `\Large`).

`\begin{hangingpars}[weite]` Alle Absätze innerhalb dieser Umgebung werden mit hängendem Einzug der Weite *weite* gesetzt. Die Angabe von *weite* ist optional. Voreingestellt sind 1,2 em.

`\Hang{vorn}txt` schreibt zuerst »vorn« hin, und dahinter mit hängendem Einzug den Rest von »txt«.

BEISPIEL FÜR HANG UND HANGINPARS

Hang Bla
Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla

Erster Hangingpars-Absatz (Weite 1,5 cm) Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla
Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla

Zweiter Absatz: Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla
Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla Bla

1.1.11. Schriftarten und -Größen

`\dunh` Umschalten auf Dunhill-Schriftart: Dunhill

Lua `\dunh` ist entfallen – wer braucht das überhaupt?

`\textbb{txt}` bzw. `\mathbb{eqn}` Diese Makros dienen zum Schreiben der »Blackboard Zeichen« $\mathbb{A} \mathbb{B} \mathbb{C} \mathbb{N} \mathbb{Z} \mathbb{Q} \mathbb{R} \mathbb{C} \dots$

Bei [palatino(OSF)] werden immer die Zeichen von pazo genommen. Sie enthalten alle Großbuchstaben (incl. einer 1).

Bei Grundschrift Computer Modern werden default die $\mathcal{A} \mathcal{M} \mathcal{S}$ -Fonts (Type1) benützt. Mit der Option [pxBBold] werden bei Grundschrift lmodern immer die `pxsyb`-Fonts (Type1) gewählt (das sind `pxfonts`). Diese sehen wie die pazo-Fonts aus und enthalten ebenfalls alle Großbuchstaben (aber keine 1).

Immer kann man mit `\mathbb{AmS}` die Blackboard-Zeichen von $\mathcal{A} \mathcal{M} \mathcal{S}$ aus dem Paket `amssym` erreichen. Obwohl meiner Meinung nach die $\mathcal{A} \mathcal{M} \mathcal{S}$ -Fonts zu viele

senkrechte Striche haben, sind sie bei lmodern-Schrift als default gewählt, weil ihr Gewicht besser zur Computer Modern passt.

In allen Varianten gibt es `\Bbbk=\mathbb{k}`.

Die Standardmengen $\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}$ und \mathbb{C} können als `\nN`, `\zZ`, `\qQ`, `\rR` und `\cC` kurz eingegeben werden. Das geht noch immer. Lua

`\textsf{txt}` bzw. `\mathsf{eqn}` schreibt fett ohne Serifen aufrecht, **also so**. Das ist nur eine Abkürzung von `\textsf{\textbf{...}}`

Hier ist `\mathsf` ein Alias für `\mathbfsfup` wegen Kompatibilität. Lua

`\mathscr{eqn}` (nur mit Option[SkriptFont])

schreibt Script-Buchstaben $\mathcal{A}\mathcal{B}\dots$ (nur Versalien).

Dieser Befehl ist eingebaut, man braucht also keine Option mehr. Lua

`\mathfrak{eqn}` Für Fraktur-Buchstaben $\mathfrak{A}\mathfrak{B}\mathfrak{C}\mathfrak{a}\mathfrak{b}\mathfrak{c}$. Verwendet Zeichen aus dem Paket `amssymb`.

Dieser eingebaute Befehl nimmt die Schrift aus `unicode XITS-math`. Lua

ANMERKUNG ZUR ZAHL DER MATH-ALPHABETE

Die folgenden eingebauten bzw. hier definierten Befehle benützen neben dem üblichen Mathematikfont eigene Math-Alphabete. In $\text{T}_{\text{E}}\text{X}$ können maximal 16 gleichzeitig verwendet werden. Diese Alphabete sind:

`mathnormal` – `mathrm` – `mathbf` – `mathit` – `mathsf` – `mathsfb`
`mathbb` – `mathscr` – `mathcal` – `mathfrak`

Dazu kommen einige »heimlich« reingesetzten vom $\mathcal{A}\mathcal{M}\mathcal{S}$ -Math. Benützt man zu viele verschiedene Math-Alphabete, so bekommt man einen Übersetzungsfehler: »too many math alphabets ...«. Hier bleibt dann nur die Möglichkeit, einige Alphabete nicht zu benützen. So kann man versuchen die Wasy-Fonts oder die Skript-Fonts nicht zu laden. Auch kann man auf den Befehl `\mathcal` zur Not verzichten und stattdessen `\CMcal` verwenden. Man kann auch `mathcal` default auf `CMcal` setzen, indem man die Option [Eucal] nicht setzt. In Palatino wird intern ein Math-Alphabet mehr benötigt, da immer noch einige Mathematische Zeichen aus den EC-Fonts genommen werden, somit spart die Verwendung von Computer Modern ein Alphabet ein.

Ein bedeutender Vorteil von $\text{Lua}\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ist, dass es keine Beschränkungen in der Zahl der Math-Fonts mehr gibt. Lua

1.1.12. Sperrung von Text in Versalien

Versalien und Kaptitälchen sollten bekanntlich immer leicht gesperrt werden.

<code>Normal</code>	=	TEXT IN VERSALIEN	und	Kapitälchen
<code>\textsc{...}</code>	=	TEXT IN VERSALIEN	UND	KAPITÄLCHEN
<code>\caps{...}</code>	=	TEXT IN VERSALIEN	UND	KAPITÄLCHEN
<code>\mcaps{...}</code>	=	TEXT IN VERSALIEN	UND	KAPITÄLCHEN
<code>\Versal{...}</code>	=	TEXT IN VERSALIEN	und	Kapitälchen

1. Das Paket `schule2e`

Es wurden dazu die Befehle `\Versal`, `\mcaps` und `\caps` so definiert, dass drei verschieden weite Sperrungen entstehen. Alle Varianten verwenden das Paket `microtype`. Im DVI-Output zeigen `\Versal`, `\mcaps` und `\caps` *keine* Wirkung.

Dabei sollte `\Versal` für Überschriften in Versalien, `\mcaps` für Überschriften mit Kapitälchen und `\caps` für Kapitälchen im Fließtext verwendet werden. `\Versal` verwendet keine Minuskelziffern, weil diese zusammen mit Großbuchstaben, wozu dieser Befehl dient, merkwürdig aussehen.

1.1.13. Spezielle Zeichen und Kombinationen

Wurzel In `schule2e` wird der `\sqrt`-Befehl aus `AMS-Math` so angepasst, dass die Wurzelexponenten etwas verschoben werden, vor allem, damit Unterlängen nicht in das Wurzelzeichen hineinreichen. Das originale Verhalten von `\sqrt` bekommt man mit `\Wurzel` wieder. Man vergleiche originale mit angepassten Wurzelexponenten:

$${}^{agg}\sqrt{a} \neq {}^{111}\sqrt{a} \quad \text{mit} \quad {}^{agg}\sqrt{a} \neq {}^{111}\sqrt{a}$$

Das in `AMS-Math` mögliche Feintuning mittels `\leftroot` und `\uproot` kann *nur* mit `\Wurzel` angewandt werden. Die Anpassung verwendet `\leftroot{-1}` und `\uproot{3}`.

\BS erzeugt einen Backslash. Wird z. B. in diesem Dokument benützt.

\eps schreibt das »schöne« ε statt des »hässlichen« `\epslilonBAD = \epsilon`. Ist ein Alias von `\varepsilon`. Auch `\epsilon` schreibt nun ε .

Var-Formen griechischer Buchstaben: Um die Eingabe griechischer Buchstaben unter AUCTex zu vereinfachen, sind für die »var-Formen« folgende Makros definiert:

$\varepsilon = \backslash\varepsilon = \backslash\eps$, $\vartheta = \backslash\vartheta$, $\varpi = \backslash\varpi$, $\varrho = \backslash\varrho$, $\varsigma = \backslash\varsigma$,
 $\varphi = \backslash\varphi$, $\varkappa = \backslash\varkappa$

Umdefinitionen: Beachte auch die fetten Varianten der undefinierten griechischen Buchstaben (bei *nicht* gesetzter Option `[orgGreek]`):

$\xi = \backslash\xi$, $\varpi = \backslash\varpi = \backslash\varpi$, $\varrho = \backslash\varrho = \backslash\varrho$.

Lua Entfällt. Alle kleinen griechischen Buchstaben sind aus `Latin Modern Math`.

\leq, \geq schreiben nun \leq, \geq , ebenso wie `\le`, `\ge`. Die Urformen \leq, \geq bekommt man mit `\leqHor`, `\geqHor`.

Lua \leq, \geq erhält man mit `Opt <`, `Opt >` auf der Mac-Tastatur.

\mEUR bzw. \peuro{8,29} bzw. \peuro[v]{11,20} Das Eurozeichen (€) kann mittels der Tastatur (mit `inputenc utf8`) direkt eingegeben werden. Es wird je nach Schriftart das Zeichen aus dem aktuellen Font (`lmodern`, `mathpazo`, `TeX-Gyre-Pagella`) genommen. Dieses ist auch als `\texteuro` eingebbar. In Lua auch mittels `\euro`.

Im mathematischen Modus kann das €-Zeichen nicht eingegeben werden, deshalb wurde noch `\mEUR` definiert, das immer eingegeben werden kann und auch vom Befehl `\peuro` verwendet wird.

Dann so: $\$ \backslash \text{peuro} \{ 12, 36 \} \$ = 12,36 \text{€}$.

`\peuro[v]{betrag}` schreibt € vor den Betrag.

`\G`, `\GZ` `\GC`, `\GF` Grad, Grad-Celsius und Grad-Fahrenheit⁴ (15° – 12,5°C – 12,2°F). Die Makros `\G` und `\GZ` sind identisch.

Da `\G` in Lua schon definiert ist, muss man stattdessen `\GZ` verwenden.

Lua

`\HMS11h5m16.26s`, `\GMS12g15m4.26s` liefert 11^h5^m16^s,26 bzw. 12° 15' 4",26. Immer müssen »h/g«, »m« und »s« sowie der ».« vorhanden sein. Fehlende Grad, Minuten, Sekunden oder Sekundenbruchteile kann man einfach weglassen, also gibt die Anweisung `\GMS g12m.s` dann 12'.

Die beiden Makros `\HMSDez` und `\HMSFig` steuern die Darstellung. Ersteres kann die Werte `{.}` oder `{,}` (default) annehmen (aber auch etwa `\cdot`) und gibt das Dezimaltrennzeichen der Sekunden an, letzteres die Werte `{l}` oder `{m}` (default) und bestimmt die Darstellung der Ziffern (lining oder minuskel).

`\MRightarrow` liefert im Textmodus das Zeichen \rightarrow .

In `marvosym` war es frech als `\Rightarrow` definiert, was schon in \LaTeX für das Zeichen \Rightarrow vergeben ist. Daher musste es undefiniert werden. In neueren Versionen von `marvosym` heißt es nun `\MVRightarrow`. Aus Kompatibilitätsgründen habe ich den Alias gelassen.

`\vH` schreibt »% « in richtigem Spacing. Ein `\xspace` ist eingebaut.

`\abk{txt}` schreibt txt mit Pünktchen nach jedem Buchstaben. Gut für `\abk{zB}` oder `\abk{oBdA}`, die z. B. bzw. o. B. d. A. liefern. Der eingefügte Zwischenraum kann vom Default `\,«` mittels

`\renewcommand{\AbkZwischenraum}{\hspace{länge}}`

beliebig angepasst werden.

`\sternchen` fügt drei Sternchen zwischen zwei Absätzen ein.

`\EnglischeLinie` fügt eine solche, $0.8 \backslash \text{linewidth}$ lang, »centered« ein. Benötigt Webo-Fonts (siehe Abschnitt 4.1 auf Seite 40)

`\SFZ` setzt das SFZ-Logo, also SFZ[®]. Schreibt man `\SFZ[v]` (steht für »verbose«), so wird ausgegeben: » Schülerforschungszentrum Südwürttemberg (SFZ[®]) «

1.1.14. Formeln und Einheiten

`\bm{eqn}` gibt die Ausgabe in »boldmath«. Muss im Math-Mode aufgerufen werden.

Ist das Original aus dem Paket `bm`. Falls es meckert, helfen oft zusätzliche

4. Das normale Grad-Zeichen kann bei `T1-fontenc latin1/utf8-inputenc` im `Textmode` direkt eingegeben werden. Es ist aber für meine Begriffe zu klein. Ebenso bei Lua.

1. Das Paket `schule2e`

Gruppenklammern.

Lua Das Paket `bm` wird nicht mehr geladen, `\bm` ist nun ein Synonym für `\mathbf` und leistet weiterhin dasselbe.

`\bra{a|b|...}`, `\ket`, `\braket` erzeugen analog die Symbole für quantenmechanische Zustände $\langle a|b\rangle$, $|a\rangle$, $\langle a|B|c\rangle$. Senkrechte Striche am Anfang bzw. Ende werden automatisch eingesetzt. Es gibt auch die Varianten mit großen Anfangsbuchstaben (`\Braket`, `\Bra`, `\Ket`), womit die Begrenzer »gestreckt« werden.

`\Pkt{a|b|c...}`, `\Set` schreibt analog $(a | \frac{2}{3} | c \dots)$ bzw. $\left\{ \sum_i (i + c) \mid 0 < |c| < 1 \dots \right\}$ mit angepasster Höhe der Klammern und Striche.

Bei `\Set` nur Anpassung des 1. Strichs. Es gibt für `Set` aus Kompatibilität zu früheren Versionen den Alias `\Menge`, sowie die kleingeschriebene Version `\set`, die nicht streckt.

`\Cdot` erzeugt einen etwas dickeren Malpunkt (\bullet).

`\D`, `\T` Abkürzung für `\displaystyle` bzw. `\textstyle`.

`\entspricht` erzeugt das Entspricht-Zeichen ($\hat{=}$)

`\Grad`, `\Div`, `\Rot`, `\ggT`, `\kgV` `\Res` Operatoren: grad, div, rot, ggT, kgV, Res.

`\re`, `\im` ersetzen die verschnörkelten \TeX -Formen » $\Re=\Re$ « und » $\Im=\Im$ « durch » \Re « und » \Im «.

`\I`, `\E` Senkrecht imaginär-»i« und Euler-»e« im Math-Modus.

`\nfrac{Zähler}{Nenner}` Brüche mit Schrägstrich z. B. $14/33$. Geht auch im Textmode. Ist nur eine Abkürzung von `\nicefrac`.

`\Ohm`, `\mur` Senkrecht Ω bzw. μ für Einheiten.

`\ov{vorn}{hint}`, `\uv{vorn}{hint}`

schreibt die »vorn«-Indexgruppe vor die »hint«-Indexgruppe. Bei `\ov` steht die »vorn«-Gruppe oben, bei `\uv` unten. Die »hint«-Gruppe jeweils auf der anderen Höhe. Das ist zum Schreiben von Tensoren gedacht.

So liefern `R\ov{i}{jkl}` und `w\ov{i}{j}\uv{k}{l}` die Ausgabe R^i_{jkl} bzw. $w^i_{jk}{}^l$. Hat man sehr viele Tensoren zu schreiben, empfiehlt sich der Einsatz des Pakets `tensind` (siehe Abschnitt 2.3 auf Seite 36), das die Eingabe sehr erleichtert.

`\pD`, `\dD` und `\iD` liefern ∂ und das senkrechte Ableitungs-»d« $f(x)dx$ im Math-Modus. `\iD` ist dasselbe wie `\dD` nur mit Half-Space davor (Integral).

`\(p)diff[n]{y}{x}` liefert die (partielle) Ableitung $\frac{d^n y}{dx^n}$ bzw. $\frac{\partial^n f(x,y)}{\partial x^n}$ auch $\frac{dy}{dx}$

`\mdiff{f}{n}{x;i;y;k;...}` erzeugt gemischte partielle Ableitungen, also so:

$$\text{mdiff}\{f\}\{6\}\{x; 2; y; 1; z; 3\} \quad \text{gibt} \quad \frac{\partial^6 f}{\partial x^2 \partial y \partial z^3}$$

`\Rpf`, `\qRpf`, `\Dpf` und `\qDpf` erzeugen » \Rightarrow «, » \Rightarrow «, » \Leftrightarrow « und » \Leftrightarrow «.

`\kurzintertext{txt}` wie \mathcal{AMS} -`\intertext`, aber mit einem kleineren vertikalen Zwischenraum. Früher hieß dieses Makro `\shortintertext`, aber das könnte einen Clash mit dem Paket `mathtools` erzeugen.

`\vorintertext{txt}` schreibt `txt` linksbündig *vor* die Gleichung, ohne deren horizontales Alignment zu ändern (sofern `txt` nicht zu lang ist). Es wird auch vertikal zentriert. Funktioniert richtig gut nur bei den Umgebungen `align` und `gather`. (Routine stammt von PHILIPP STEPHANI). In älteren Versionen von `schule2e` hieß dieses Makro noch `\inlineintertext`. Auch `aligned` geht, allerdings mit Verrenkungen, wie in diesem Beispiel:

```
\begin{gather}
  \beforetext{\$displaystyle%
    \begin{aligned}&!\text{und ebenso}\%
      &!\text{Also}\end{aligned}\$}
  \begin{aligned}
    hwnn &= H+hw\\
    nn-nn &= 1.
  \end{aligned}
\end{gather}
```

Die Klassenoption `leqno`, die Gleichungsnummern links setzt, arbeitet mit diesem Befehl nicht zusammen.

`\starlabel{label}` dient zur Einfügung von automatischen Gleichungsnummern und Labeln in Sternformen von `amsmath`. Der `label` kann mit `reflex` erzeugt werden.

`\str{PQ}`, `\Str{PQ}` und `\Pvek{PQ}` erzeugen der Reihe nach Ausgaben wie AVB , $\overline{A_1B_1}$ bzw. Vektorpfeile $\overrightarrow{P_1Q}$. Dabei wurde das Spacing zwischen den Buchstaben entfernt, weil die Zeichen nun in `\mathit` gesetzt werden. Vergleiche $\$AVB\$ = AVB$ mit $\str{AVB} = AVB$.

Da in `\mathit` griechische Buchstaben nicht in allen Fonts (etwa `mathpazo`) erreichbar sind, wurden noch die Makros `\tstr`, `\tStr` und `\tPvek` definiert, die das Argument in `\text{...}` packen. Dabei wird allerdings wieder ein Spacing wie bei $\$AVB\$$ erzeugt.

Das entspricht z. B. $\Str{AB\text{\$\Gamma\$}C}=\overline{AB\overline{C}}$

Hier wird in `\textit` gesetzt. Die `\t...-Befehle` sind dasselbe. Man *muss* griechisch mit der Tastatur und Potenzen bzw. Indices mit `\TO` bzw. `\TU` eingeben. »Striche« mit `\p`, `\pp`.

Lua

`\TO(*){txt}`, `\TU(*){txt}`, `\p`, `\pp` sind Analoga für $\hat{\text{txt}}$, $_{\text{txt}}$, sowie $'$ und $''$ für den Text-Mode. In *Sternformen* setzen sie `txt` italic, sonst aufrecht.

`\U{einh}`, `\Ut{einh}` Zum Setzen von Einheiten, ggf. unter Verwendung von `\frac`, falls ein \gg/\ll in der Einheit auftaucht, z. B. 12 kgm/s^2 , $18 \mu\text{F}$ und $11 / \text{mol}$,

1. Das Paket schule2e

aber 12m^2 . Potenzen von Einheiten können nur im Math-Modus eingegeben werden. Im Textmodus wird die aktive Schriftart verwendet, etwa » $1,23\text{m/s}$ «, im Mathmodus immer \mathrm außer in Captions, die ja default \mathrm benutzen, dort werden alle Einheiten in \mathsf geschrieben, so man das Makro \mathsf nicht umdefiniert hat. Kommen Potenzen von Einheiten vor, so muss man natürlich immer den Math-Modus verwenden. Die Befehle \U und \Ut sind identisch.

Lua Da \U in Lua schon definiert ist, muss man stattdessen \Ut verwenden.

Es gibt folgende Optionen:

- [b] Vs/m^2 das ist der Default »Bruch«
- [f] $\frac{\text{Vs}}{\text{m}^2}$ »frac«
- [x] Vs/m^2 kann auch jeder sonstige Buchstabe sein

$\text{Spvek}[A]\{a; b; \dots\}$ erzeugt Spaltenvektoren. Der optionale Parameter A steht default auf [r], also für rechtsbündige Ausrichtung. Alternativen: [c] für zentriert und [l] für linksbündig. Die Zeilen werden bei der Eingabe durch »;« abgetrennt.

vek Der Befehl zum Setzen von Vektoren. In der Voreinstellung werden sie italic fett gesetzt. Beachte auch die Optionen [Pfeilvek] und [Gothicvek], falls Pfeile bzw. Fraktur gewünscht werden.

evek Erzeugt die »amerikanischen« Einheitsvektoren, die ein Dach haben, also $\hat{a}, \hat{\omega}$. Beachte auch die unterschiedliche Darstellung bei den Optionen [Pfeilvek] und [Gothicvek].

Op Zum Schreiben von Operatoren und Tensoren. Darstellung ist gewöhnlich sans-serif fett aufrecht. Makros, etwa griechische Buchstaben, sind normal fett:
 $\mathbf{B} = \Phi^{-1} \mathbf{A} \mathbf{H}$, auch: Φ (mit \upPhi).

Lua Hier werden alle Zeichen »von selbst« sans-serif-fett und aufrecht, auch die Griechischen.

IM $\text{IM} = \mathbb{1}$ ist die Einheitsmatrix.

Lua Winkel erzeugt das Winkelzeichen » \sphericalangle «. Ist sphericalangle aus amsfonts oder unicode-math.

$\text{Z}\{x\}$ Schreibt » $\cdot 10^x$ «.

oiint liefert ein »selbstgebasteltes« Oberflächenintegral, sofern nicht die Option [WasyInt] verwendet wird. So sieht es aus:

$$\oiint D \, dA = Q$$

Lua Integrale gibt es nahezu alle vorstellbaren, auch dieses (oiint).

cupdot ist das Zeichen für disjunkte Vereinigung $A \cup B$.

bigcupdot ist das entsprechende Operatorsymbol $S = \bigcup_{i=1}^n A_i$ bzw. $S = \bigcup_{i=1}^n A_i$

Lua Diese beiden Zeichen existieren in unicode-math unter denselben Namen.

1.1.15. Sonstige Makros

- `\ArrF{fakt}` erhöht den Zeilenabstand in Arrays und Tabellen um das *fakt*-fache.⁵ Standardwert ist 1.
- `\Fixdate<tag>.<monat>.<jahr>`. setzt Tag, Monat und Jahr für folgende Datumsangaben mit `\today`. Man beachte die Notwendigkeit **aller drei Punkte**, auch dem hinter der Jahreszahl.
- `\qbox{txt}` und `\qqbox{txt}` ist `\quad\text{...}\quad` bzw. `\qqquad ...`
- `\Setzbreiten{a}` setzt die Längenvariable `\Breite` auf den Wert *a* und die Längenvariable `\breite` auf »`\linewidth - a - 1 em`«. (gut für nebeneinanderstehende Minipages!)
- `\Sumbreite{a}{b}` setzt die Längenvariable `\breite` auf *a + b*.
- `\Sr{y}{h}` erzeugt eine »Strut« also eine (unsichtbare) rule der Breite 0pt, der Höhe *h ex*, die *y ex* gegenüber der Grundlinie verschoben ist.
- `\tabcaption{txt}` und `\figcaption{txt}` erzeugen eine caption außerhalb der Umgebungen `table` bzw. `figure`, die aber konsistent mit den `figures` bzw. `tables` weitergezählt wird.
Ein `\label`-Befehl muss innerhalb der `{...}` von `fig/tabcaption` stehen. Auch ein optionaler Parameter für das Verzeichnis (wie bei `\caption[...]`) funktioniert. Allerdings darf hier der optionale Parameter nicht mit einem Punkt beginnen. Aber wer macht das schon.
- `\TabV{txt}` richtet genügend vertikalen Platz für `txt` in einer Tabelle ein.
- `\Uhr{12}{18}`, `\Uhr[o]{12}{18}` Schreiben »12¹⁸ Uhr« bzw. »12¹⁸ Uhr« (oldstyle).
- `\todayInter` schreibt das Datum in internationaler Form: 2013-02-17.

1.1.16. Tabellenbearbeitung

Zur Tabellenverarbeitung werden die Pakete `booktabs`, `array`, `numprint` (mit Option `np`), `tabularx` geladen, sowie `rotating`, welches seinerseits `graphicx` lädt, dann werden eine Reihe von Makros und Spaltenstilen definiert. (vgl. Tab. 1.4 auf der nächsten Seite)

Bei den Stilen `P`, `M`, `B` und `T` ist die Paragraphenbreite anzugeben, z. B. `P{3 cm}`.

In früheren Versionen von `schule2e` hatte ich eine Umgebung **oldstyletabellen** verwendet, die es nun wegen der Kompatibilität zu `LuaLATEX` **nicht** mehr gibt. Ihr Sinn war, Tabellenziffern am Dezimalpunkt ausgerichtet in Minuskeln setzen zu können.

Nun sollte man so vorgehen: Will man Minuskel, nehme man den Spaltenstil »`N`«, will man Math-Mode Ziffern, nehme man den Stil »`n`« (beide aus `numprint`).

Der Spaltenstil »`N{v}{n}`« aus `numprint` richtet am Dezimaltrenner aus und setzt Ziffern im Text-Font mit Tausender-Trennern (somit in `schule2e` in Minuskelziffern).

5. Dieser Befehl muss außerhalb des Arrays gesetzt werden. Tritt er innerhalb einer Umgebung z.B. einem `{...}` oder `\left(... \right)` auf, so setzt er sich bei Verlassen (wie üblich) zurück.

TAB. 1.4: Vordefinierte Spaltenstile

Beschreibung	Ausrichtung					
	c	l	r	p	m	b
Flattersatz-Paragraph	-	-	-	P	M	B
um 90° gedreht in small	-	-	-	T	-	-
um beliebige ° gedreht in small	-	-	-	G	-	-
Ausrichtung am Dezimaltrenner	n, N					
row-Style ändern	+, =					

Der analoge Stil » $n{v}{n}$ « setzt im Math-Font.

Beispiel zur Verwendung des Stils »T« (gedacht) für eine Spaltenüberschrift:

```
\multicolumn{1}{T{6em}}{laber, laber, ...}
```

Die Spaltenüberschrift wird dann 6em »hoch«.

Analog bei z. B. 45° nehme man G statt T in der Form $G{45}{6em}$ o. ä.

Auch mehrzeilige Einträge gehen mit T bzw. G, denn der Text kommt in eine minipage.

ANMERKUNG ZU NUMPRINT

Das Paket wird mit der Option `[np, boldmath]` geladen und mittels `\npdecimalsign{.}` der Punkt als Dezimaltrenner bei der Ausgabe gewählt. Für die Eingabe kann man Punkt oder Komma verwenden. Dieses Paket stellt den Spaltenstil

```
N/n{vorkomma}{nachkomma}
```

zur Verfügung. Der Hauptbefehl dieses Pakets ist `\np[Einheit]{zahl}`. Dabei darf `zahl` die Zeichen » $+-\pm 0123456789eEdD$ « enthalten. Ausgegeben wird dann mit Tausender-Trennern und mit »e, E, d« oder »D« erscheinen 10-er Exponenten.

»N« setzt im Text-Font und »n« im Math-Font.

Schalter für die Ausgabe (mit meinen defaults) sind `\npdecimalsign{.}`, `\npthousandsep{\,}` und `\npproductsign{\ensuremath{\cdot}}`.

`\npboldtext` Das ist das Analogon zu `\npboldmath`. Dieser Befehl schreibt Tabelleneinträge in bold. Man muss nur `{\npboldtext}` für Minuskelfziffern (n) und `{\npboldmath}` für Versalziffern (N) vor den Tabelleneintrag setzen.

Lua Beim Stil n wird bei `\npboldmath` der Malpunkt vor den Zehnerpotenzen nicht angezeigt.

STILE DER ZEILEN EINER TABELLE ÄNDERN

Folgendes Beispiel zeigt, wie man für ganze Zeilen den Style ändern kann. Von `schule2e` wird dazu das Makro `\zeilstyle` und die Columntypes `'='` und `'+'` definiert. `'='` nimmt den `zeilstyle` von der Zelle, `'+'` fügt ihn der Zelle hinzu. Die erste Zelle sollte mit `'='` beginnen, alle anderen auf die `zeilstyle` wirken soll mit `'+'`. `\zeilstyle` setzt den Stil für die folgende Zeile. Setzt man `\zeilstyle` *nicht* vor die erste Spalte, werden die folgenden Spalten der Zeile hervorgehoben.

Eine Anwendung geht dann z. B. so:

```

\begin{tabular}{ =1 | +1 +1 1 +1 }
  \zeilstyle{\color{red}}
    & 1 & 2 & 3 & 4 \\ \hline
1 & a & b & & 
  \zeilstyle{\color{green}} c & d \\
2 & a & b & c & d \\
\zeilstyle{\Large\bfseries\itshape}
3 & a & b & c & d \\
4 & a & b & c & d \\
\end{tabular}

```

mit der Ausgabe:

	1	2	3	4
1	a	b	c	d
2	a	b	c	d
3	<i>a</i>	<i>b</i>	c	<i>d</i>
4	a	b	d	d

Für Blöcke, Randbeschriftungen, »bordermatix« usw. beachte man das Paket `blkarray`.

HILFSFUNKTIONEN FÜR TABELLEN

`\keinPlatz` ist ein Objekt, das zwar vorhanden ist, aber keinen Platz belegt. Nett zur vertikalen Ausrichtung. Auch `\null` ist hier oft nützlich.

`\BildTop{bild}` setzt `bild` in eine `\vbox` mit `\null` als erste Zeile und darunter `bild` in eine `\hbox`. Dadurch wird in Tabellen das Bild auf die erste Zeile ausgerichtet. Man kann auch anderes als ein Bild in `\BildTop` verpacken.

`\TabBild{spalten}{weite}{bild}` ist eine Abkürzung von:

```

\multicolumn{spalten}{p{weite}}{
  \BildTop{\includegraphics[width=weite]{bild}}
}

```

`\MRzell[verticalign]{horalign}{stuff}` erzeugt eine Tabellenzelle in der ein Zeilenumbruch mittels `»\«` erlaubt ist. Intern ist dieser Eintrag eine Tabelle. Dabei ist `verticalign` die vertikale Ausrichtung der Tabelle, also `c` (default), `t` oder `b`. `horalign` die horizontale Ausrichtung, also `c`, `l` oder `r`. Dieser Befehl ist gedacht für Zellen des Typs `c`, `l` oder `r`. Die Zelle wird dann so breit wie die längste Zeile von `stuff`. Ein Beispiel ist Tab. [1.5 auf der nächsten Seite](#).

Um mehrere Tabellenzeilen zu überspannen, wurde das Paket `multirow` integriert. Dieses stellt folgenden Befehl zur Verfügung (mehr dazu in der Anleitung):

`\multirow{zeilen}{weite}[fixup]{stuff}` wobei `zeilen` die Anzahl der zu überspannenden Zeilen (negativ »nach oben«), `weite` die horizontale Breite der

1. Das Paket schule2e

TAB. 1.5: In dieser Tabelle mit `{|c|c|c|}` wurde die zweite Zelle mit `\MRzell [b]{r}{FooFoo\bar}` gesetzt.

Foo	FooFoo bar	FooFoo Foo
-----	---------------	------------

Zellen und stuff der Zelleninhalt ist. Setzt man `{weite}` auf `{*}`, so wird die natürliche Breite von `stuff` verwendet, sonst kommt er in eine `\parbox`. `fixup` ist eine Länge zum Finetuning: Positive Werte schieben `stuff` nach oben, negative nach unten. Default wird `stuff` vertikal zentriert.

Ein Beispiel für `\multirow` ist:

Dies	Das	Jenes	Sonst
1	Tri Tra Trallala, wo ist		b
2	denn der böse Wolf?		c
3	Das sollte man heraus-		d
4	finden, damit einem		e
5	nichts passiert.		f
6	Gurke	Salat	g

Diese Tabelle wurde mit dem Code in Abb. 1.1 gesetzt.

```

\begin{tabular}{@{}l|l|l|c@{}}
Dies & Das & Jenes & Sonst \\
\midrule
1 & \multirow{5}{4cm}{\raggedhyphen Tri Tra Trallala,
    wo ist denn der böse Wolf?
    Das sollte man herausfinden,
    damit einem nichts passiert.} & & b \\
2 & & & c \\
3 & & & d \\
4 & & & e \\
5 & & & f \\
\midrule
6 & Gurke & Salat & g \\
\bottomrule
\end{tabular}

```

ABB. 1.1: Code der multirow Tabelle

2. Weitere Pakete

2.1. Das Paket `typeareaB`

Der `typeareaB`-Stil ist der Stil aus dem Vorläufer des KOMA-Pakets. Es wird folgender Befehl definiert: `typearea[BCOR]{n}` (BCOR ist der Binderand.)

Er teilt Höhe und Breite der Seite in je n gleiche Teile. Dadurch entstehen horizontale und vertikale Längeneinheiten HLE und VLE. Bei `twoside`-Druck werden nun der innere Rand auf 1 HLE, der äußere Rand auf 2 HLE gesetzt, bei einseitigem Druck beide Ränder auf 1,5 HLE. Stets wird ein oberer Rand von 1 VLE und ein unterer Rand von 2 VLE freigelassen. Der Seitenkopf liegt innerhalb des oberen Rands. Randbemerkungen stehen im äußeren Rand und haben eine Breite von 1,5 HLE bei `twoside`-Druck und 1 HLE bei einseitigem Druck. Standardmäßig wird bei einer Schrift von 10 pt, 11 pt bzw. 12 pt der Teiler n zu 8, 10, bzw. 12 gewählt. Kopf und Fuß gehören nicht zum Textbereich. Soll das Werk gelocht oder in einen Klemmhefter gesteckt werden, wodurch dann der innere Rand verkleinert wird, so sollte man als Option eine Länge an `schule2e` mitgeben (12 mm haben sich bewährt), wodurch dann `\typearea[<Länge>]{...}` aufgerufen wird.¹ Somit ergeben sich etwa für die A4-Seite die Maße der Tabelle 2.1 auf der nächsten Seite. Die Höhen stimmen nicht exakt, da sie von `typearea` in Vielfachen von `\baselineskip` berechnet werden.

Einen anderen Satzspiegel kann man mittels `\typearea[<Länge>]{<Teiler>}` festlegen. Dabei ist `<Teiler>` der oben erwähnte Teiler, und `<Länge>` ist die Bindungskorrektur; ist sie (mit Maßeinheit!) angegeben, so wird die Seite als um diesen Betrag schmaler betrachtet, und auf den »Rest« die Einteilung in `DIV` Zellen angewendet, und schließlich der innere Rand um diesen Wert vergrößert. Beachte auch die spezielle Setzung von **111 in** für diese Länge (siehe Option Binderand 1.1.1 auf Seite 7)

Die Größe des Papiers kann ebenfalls übergeben werden, dazu ist in der Deklaration von `\documentclass` eine der Papiergrößen-Optionen aus Tabelle 2.2 auf der nächsten Seite anzugeben.

Daneben existiert noch der Befehl `\areaset[BCOR]{Breite}{Höhe}` der die Textbereichsbreite und -höhe vorgibt und ein Ränderverhältnis wie bei `\typearea` einstellt, d.h. die horizontalen Ränder haben das Verhältnis 1:2 bzw. 1:1 beim einseitigen Druck und die vertikalen Ränder bekommen das Verhältnis 1:2.

1. Diese Seitenaufteilungsvorschrift ist die von Tschichold als besonders gut empfohlene.

2. Weitere Pakete

Tab. 2.1: Seitenmaße in mm

n	Textbereich		Ränder			
	Breite	Höhe	oben	unten	eins. innen	zweiseitig außen
8	131.25	185.62	37.12	74.25	39.37	44.62
10	147	207.9	29.7	59.4	31.5	35.7
12	157.5	222.75	24.75	49.5	26.25	29.75

Tab. 2.2: Papiermaße in mm, Zoll und PostScript-Punkten

Papieroption	europäisch [mm]/[bp]		Papieroption	Englisch [in]/[bp]	
	Höhe	Breite		Höhe	Breite
a4paper	297/842	210/596	letterpaper	11 / 792	8.5 / 612
a5paper	210/596	148/421	legalpaper	14 / 1008	8.5 / 612
a6paper	148/421	105/298	executivpaper	10.5 / 756	7.25 / 522
b5paper	250/711	176/504			

Bei Wahl des Papierformats [a5paper] als `\documentclass`-Option wird default der Teiler 11 gewählt. Hier sollte auch die Schriftgröße auf 10pt gesetzt werden. Damit bekommt man eine Textbreite von 108 mm.

An diese Option kann noch [landscape] (am besten schon in `\documentclass`) angeschlossen werden, was eine Vertauschung von Höhe und Breite bewirkt. Beliebige Größen und Randeinstellungen gehen mit dem `ansize`-Stil, allerdings werden hierbei die Randbemerkungen nicht mit eingestellt. Dies leistet z. B. der `geometry`-Stil.

Will man `typeareaB` ohne `schule2e` benutzen, so muss man die Länge `\brfBCOR` vor der Einbindung definieren und festlegen; sie bestimmt die Defaulteinstellung für die Bindungskorrektur. Es ist dann allerdings vielleicht besser, gleich das originale Paket `typearea` aus dem KOMA-Bundle zu nehmen.

2.2. Die Klasse `briefbrf.cls`

Diese Klasse ist eine Anpassung des alten `script_l`-Stils von Frank Neukam auf $\text{\LaTeX} 2_{\epsilon}$. Manche Felder haben je nach Sprache (german, english (default) bzw. french) andere Vordrucknamen. So wird z.B. mittels `\yourmail` ein Referenzfeld mit dem Vordruck »Ihr Brief vom« gesetzt. In Englisch wäre dies »your letter of« usw. Dieser Vordruckname kann mittels `\yourmailname` geändert werden. Dies trifft analog für alle sprachspezifischen Vordrucknamen zu. Die Liste der Befehle findet sich in [Tabelle 2.3 auf der nächsten Seite](#). Die Klasse lädt selbst `babel`.

TAB. 2.3: Befehle und Felder der Klasse `briefbrf.cls`

<code>\begin{letter}{Anschrift}</code> beginnt einen neuen Brief.	<code>\yourmail{Ihr Schreiben vom}</code>
<code>\end{letter}</code> Ende des Briefs.	<code>\myref{unser Zeichen}</code>
<code>\opening{Anrede}</code> setzt alle Teile des Briefs vor der Anrede.	<code>\customer{Kundennummer}</code>
<code>\closing{Gruß}</code> setzt die Grußformel und die Unterschrift.	<code>\invoice{Rechnungsnummer}</code>
<code>\ps{blabla}</code> setzt ein Postscriptum.	<code>\refitemi{Eigenes Feld 1}</code>
<code>\cc{Verteiler}</code> Verteilerliste.	<code>\refitemii{Eigenes Feld 2}</code>
<code>\encl{Anlagen}</code> eben diese.	<code>\refitemiii{Eigenes Feld 3}</code>
<code>\addrheight</code> , <code>\addrwidth</code> und <code>\addrvskip</code> , <code>\addrindent</code> legen die Position und Ausdehnung des Adressfeldes fest.	Die »Vordrucke« zu den letzten Feldern werden mit <code>\refitemnamei</code> usw. eingestellt.
<code>\backadress{Absender}</code> legt eine Absenderzeile ins Adressfeld.	<code>\subject{blabla}</code> schreibt das Betreffsfeld, mit <code>\subjecton</code> kann man das Wort »Betr.« voranstellen lassen. Das voreingestellte <code>\subjectoff</code> schaltet dies aus. Analog schaltet <code>\subjectbold(on/off)</code> den Fettdruck der Betreffzeile (default off).
<code>\specialmail{Versandart}</code> blendet eine Versandart (z.B. Einschreiben) ins Adressfeld ein.	<code>\title{Überschrift}</code> erzeugt unterhalb des Referenzfelds eine Überschrift.
<code>\location{Information}</code> erlaubt Eintragung weiterer Informationen rechts neben dem Adressfeld in ein <code>\locwidth</code> breites rechtsbündiges Feld.	<code>\firsthead{Kopfzeile}</code> und <code>\firstfoot{Fußzeile}</code> bestimmen die Kopf- und Fußzeile der ersten Seite, so dass auf diesen der Absender, die Bankverbindung usw. eingetragen werden kann.
Referenzfeld In diesem Feld stehen, sofern aufgeführt, die folgenden Felder:	<code>\foldmarks</code> und <code>angehängtes on bzw. off</code> schalten die Falzmarkierungen, die default eingeschaltet sind.
<code>\yourref{Ihr Zeichen}</code>	

2. Weitere Pakete

schule2e definiert auch den Befehl `\BitteWenden`, der unten rechts auf der aktuellen Seite die Floskel – bitte wenden – einfügt.

SERIENBRIEFE

Serienbriefe können auf die folgende Art und Weise erzeugt werden:

In eine ASCII-Datei z.B. »serie.adr« schreibt man Einträge der Form:

`\ADR{Feld1}{Feld2}{Feld3}{Feld4}{Feld5}{Feld6}{Feld7}{Feld8}{Feld9}`

wobei die Felder z.B. mit Name, Vorname, Anschrift, ... belegt sein können. Dann startet man den Serienbrief wie im Code-Schnipsel (Abb. 2.1) angegeben. Die Zahl der Felder kann natürlich auch geringer aber nicht größer sein.

```
\documentclass{letter}
\begin{document}
\newcommand{\ADR}[9]{%
  \begin{letter}{#2 \#1\#3}
  \opening{Liebe #1,}
  ...
  \end{letter}
}
\input{serie.adr}
\end{document}
```

ABB. 2.1: Code für Serienbrief

Diese Technik funktioniert entsprechend für alle Serierendokumente. Mein Python-Skript `CSVtoADR.py` kann solche Dateien aus einer `.csv` Datei erzeugen.

2.3. Das Paket `tensindbrf`

Dieses Paket ruft das Paket `tensind` auf und setzt gleich mir genehme Parameter. So wird das »?« als Begrenzer für Tensoren gesetzt und als Format »lrb« gesetzt, was Pünktchen in der unteren Index-Reihe bewirkt, falls dort Leerstellen sind. Vor den Tensorableitungen »,<« und »;<« wird ein kleiner Leerraum eingefügt und danach werden keine Pünktchen mehr gesetzt.

Die Eingabe eines Tensors geht mit diesem Paket ganz einfach: Die Eingabe von $?R^i_{jkl}?$ liefert das erwartete R^i_{jkl} . Normalerweise werden in der unteren Indexreihe für Leerstellen Pünktchen eingesetzt. Mit $?[]R^i_{jkl}?$ werden diese nicht geschrieben. Statt `[]` kann auch z. B. `[n]` geschrieben werden.

2.4. Das Paket `palateuler`

Dieses Paket verwendet Palatino, Myriad (serifenlos) und `lmtt-super`-Fonts für Typewriter. Die Größen werden auf die Palatino angepasst. Für den Mathematik-Satz werden

die *mathpazo*-Fonts benützt. Früher waren die Euler-Fonts benützt worden, daher der »unpassende« Name. Das Paket *mathpazo* wird zu Beginn geladen. Es setzt u. a. griechische Großbuchstaben schräg (wie es sich eigentlich gehört!) und besitzt eigene *boldmath*-Fonts. Diese Schriftkombination liefert hervorragende Resultate, auch als PDF. Es wird von *schule2e* geladen, wenn die Option *palatino* (oder *palatinoOSF*) gesetzt ist.

Das Paket *palateuler* kennt die beiden Optionen:

pplj Defaulteinstellung. Sie verwendet die (gekaufte) Palatino-SC&OSF für durchgängige Minuskelziffern im Text-Modus und echte Small-Caps.

Ab *teTeX* 3.0 sind nun freie Varianten von SC&OSF vorhanden. Um es zu aktivieren, muss nur die *fp1s.map* in die *updmap* aufgenommen werden. Siehe für die sonstigen notwendigen Files die Datei *INSTALL2E* auf meiner Homepage. In *texlive2010* war *fp1s.map* schon default in der *updmap* drin.

ppl Hier werden »gefakte« Small-Caps und keine Minuskelziffern als Standard verwendet. Diese Option ist zu verwenden, wenn man die Palatino-SC&OSF nicht besitzt, was nur noch auf sehr alten *T_EX*-Distributionen der Fall sein sollte.

Man beachte, dass bei Erzeugung eines PDF-Files bei beiden Optionen der Schalter *-G0* beim *dvips*-Lauf zu setzen ist. Dies macht das unten beschriebene Skript *dvi2pdf* automatisch.

EC-FONTS UND PDF

Ein PDF mit EC-Fonts ist indiskutabel, sofern man keine *cm-Super* Fonts installiert hat. Mit diesen bekommt man aber hervorragende PDF-Dateien mit Computer Modern. Wenn man schon dabei ist, sollte man mit seinem Lieblingseditor die Datei *cm-super-t1.enc* öffnen und in den letzten Zeilen sicherstellen, dass sie so aussehen:

```
%/germandbls % or
/germandbls.alt
] def
```

Damit bekommt man das schönere, alte »ß« wie in den *cm*-Fonts, anstelle des gequetschten aus den *ec*-Fonts.

Eine gute Alternative für diejenigen, die keine 50 MB *cm-Super*-Fonts runterladen möchten, können die neuen *lmodern* Fonts sein. In der größten Not kann man die virtuellen *ae*-Fonts benützen, die Type1-Schriften liefern. *Inzwischen verwendet schuleze die lmodern Fonts.*

3. Externe Hilfsprogramme

3.1. Eigene Routinen

3.1.1. Booklet und DviDuplex

Booklet ist ein Hilfsprogramm, um zwei Seiten auf ein A4 Blatt zu drucken oder A5-Heftchen herzustellen. Es verwendet das Skript `psopts` von JOSEPH HARRINGTON. Sein Aufruf erfolgt über die Kommandozeile:

```
booklet [ -h -2 -s -p -m -C ] Dateiname (ohne Extension)
```

Ohne Option oder Dateiname wird nur die usage ausgegeben. Es kann vom AUCTeX Command Menü direkt aufgerufen werden. Die Optionen bedeuten:

- h Hilfetext wird angezeigt
- 2 Die Bögen sollen nur einseitig bedruckt werden; default ist Druck auf Vorder- und Rückseite zur Erstellung von Heftchen.
- s Postscript soll die Seite verkleinern. Normalerweise sollte die dvi-Datei schon Seiten im A5-Format erzeugt haben. Das ist eher für nicht selbst erstellte Dokumente gedacht.
- p Die Datei liegt schon im Postscript-Format vor. Es ist also kein `dvips`-Lauf durchzuführen. Gut brauchbar, wenn nur einzelne Seiten gedruckt werden sollen.
- m Das Quell-dvi ist aus einem 12-pt A4-Dokument durch Verkleinerung mit `\mag 707` entstanden. Dazu müssen die Seiten leicht auf dem »Zielbogen« verschoben werden. Dieser Schalter sollte nicht zusammen mit `-s` verwendet werden.
- C *n* Druckt *n* Kopien des Booklets.

Inzwischen ist meine neue Version auf meinen Duplex-Drucker zugeschnitten. Die im Netz auch veröffentlichte Version `booklet-man` geht vom Druck auf Simplex-Drucker aus, bei dem man also Vorder- und Hinterseiten einzeln drucken muss. Dort heißt der Schalter `-m` noch `-k` und `-C` existiert dort (noch) nicht, weil es auch wenig Sinn macht, da man ja den Papierstapel von Hand umdrehen muss.

Mit **dviduplex** kann man eine `.dvi` oder `.ps` Datei auf einem Duplex-Drucker (Vorder- und Rückseite können bedruckt werden) ausgeben lassen. Aufruf ist:

```
dviduplex [ -h -s ] Dateiname (ohne Extension)
```

Die .dvi-Datei wird mittels `dvips` in eine .ps-Datei übersetzt und dann mittels `psopts` an den Drucker zur Ausgabe im duplex Modus geschickt. Die Option `»-s«` (skip) unterdrückt den Lauf `dvi→ps`, falls die .ps Datei schon vorhanden ist.

3.1.2. Dvi2Pdf

Wie der Name schon andeutet, wandelt dieses Programm eine .dvi-Datei in eine .pdf-Datei um. Der Aufruf lautet einfach

```
dvi2pdf [Option] Dateiname (ohne Extension)
```

Programm-Ablauf: Zuerst wird `dvips -G0 -Pps Dateiname` durchgeführt. So wird sichergestellt, dass nur Postscript-Fonts verwendet werden und wegen des Schalters `-G0` werden Fehler wie `»£«` statt `»fi«` ausgebügelt.

Anschließend wird die entstandene Postscript-Datei durch `ps2pdf` von GhostScript bzw. den Acrobat-Distiller geschickt. Es gibt folgende Optionen, die das Format der PDF-Datei bestimmen:

- d Benütze Acrobat Distiller zu Konvertierung. Das geht bisher nur auf Mac, mittels eines Shell-Scripts und eines Apple-Scripts. Ansonsten wird `ps2pdf 14` aus Ghostscript zur Erzeugung eines pdf für Acrobat 5 und höher benützt.
- s Unterdrücke (skip) die Erstellung des ps-files. Dies macht nur dann Sinn, wenn vorher schon ein aktuelles .ps erzeugt wurde.
- h Eine Hilfe anzeigen.

Hat man sein Dokument mit Computer Modern und Eco-Fonts gesetzt, dann scheint `ps2pdf` die Ziffern im Suchfenster von Apples Preview und Acrobat (nicht im Text!) als `»wundersame«` Zeichen darzustellen. Eine Konvertierung der ps-Datei mit Acrobat Distiller scheint dagegen sauber zu sein.

4. Hinweise zu Paketen

4.1. Webo-Fonts

Die auf CTAN erhältlichen Fonts `WeboMintsGD` stellen eine Reihe sehr hübscher Ornamente zum Umrahmen oder Trennen von Absätzen zur Verfügung.



Das hier ist so eines, horizontal 8-fach gestreckt und zentriert. Man kann sich ein Makro schreiben, um solch ein Zeichen auszugeben:

```
\newcommand*\WeboZeichen[3]{\%  
  \fontsize{#1}{#2}\usefont{U}{webo}{x1}{n}#3}}
```

Es sind der Reihe nach die Parameter Fontgröße und -Durchschuss sowie das Symbol anzugeben. Eine Liste der vorhandenen Symbole enthält die Tabelle [4.1 auf der nächsten Seite](#). Das obige Symbol wurde so erzeugt:

```
\begin{center}  
  \scalebox{8}[1]{\WeboZeichen{10}{12}{4}}  
\end{center}
```

Das ist die `\EnglischeLinie` zur Absatztrennung, nach der – wie nach einer Überschrift – nicht eingerückt wird:




































































Hier geht's weiter ...

4.2. auto-pst-pdf

Häufig möchte man seinen Quelltext direkt nach pdf übersetzen. Das geht meist ohne Mühe, aber im pdf-mode kann \LaTeX nicht mit `pstricks` oder `psfrag` umgehen. Dieses Paket von WILL ROBERTSON und JOHANNES GROSSE schafft Abhilfe, indem es einen ersten Lauf initiiert, der alle Bilder, `pspicture`-, `postscript`- und `psfrag`-Umgebungen durch `tex`→`dvi`→`ps`→`pdf` jagt. Die Ergebnisse werden in Hilfsdateien

TAB. 4.1: Ornamente aus WeboMintsGD

/ → 	o → 	1 → 	2 → 	3 → 	4 → 
5 → 	6 → 	7 → 	8 → 	9 → 	
A → 	B → 	C → 	D → 	E → 	F → 
G → 	H → 	I → 	J → 	K → 	L → 
M → 	N → 	O → 	P → 	Q → 	R → 
S → 	T → 	U → 	V → 	W → 	X → 
Y → 	Z → 	[→ ] → 		
a → 	b → 	c → 	d → 	e → 	f → 
g → 	h → 	i → 	j → 	k → 	l → 
m → 	n → 	o → 	p → 	q → 	r → 
s → 	t → 	u → 	v → 	w → 	x → 
y → 	z → 				

abgelegt und dann wird ein normaler pdf-Lauf durchgeführt. Hat man einmal alle Bilder bearbeitet, kann man dem Paket die Option [off] mitgeben, wodurch der oben beschriebene Bildbearbeitungslauf nicht durchgeführt wird.

Übersetzt man mit Ziel dvi, so macht das Paket nichts, allerdings funktionieren die von ihm definierten Makros `\mathfig` und `\psfragfig` weiterhin.

Das Paket lädt die Pakete `ifplatform`, `pst-pdf`, `xkeyval`. Das Perl-Script `pdfcrop` muss installiert sein (ist es üblicherweise).

`pdflatex` muss mit der Option »-shell-escape« aufgerufen werden.

Ebenso `lualatex`.

Lua

Nötig ist das Laden des Pakets nur noch bei Verwendung von `pstricks` oder `psfrag`. Heutiges \LaTeX wandelt alle eps Bilder automatisch unter Verwendung von `epstopdf` ins pdf Format um und fügt die Endung `eps-converted-to.pdf` an.

A. schule2eLua

schule2eLua ist die auf Lua \LaTeX angepasste Variante von schule2e. Es werden otf-Fonts und das Paket `unicode-math` eingesetzt.

Verwendete Schriften sind:

`TeX Gyre Pagella` Grundschrift (Palatino-Klon) für roman.

`MyriadPro` Serifenlose Schrift. Mit Option `[helv]` wird stattdessen `TeX Gyre Heros` genommen.

`DejaVu Sans Mono` Monospaced Schrift.

`TeX Gyre Pagella Math` Standard Math Font.

`XITS-Math` `\mathcal` und `\mathscr`.

`Asana-Math` `\mathbb`

`Latin Modern Math` griechische Kleinbuchstaben.

Der Vorteil von Lua ist, dass alle Unicode-Zeichen direkt von der Tastatur aus in den Quelltext eingegeben werden können. In Emacs kann man die Nummern mittels »F4« reinklopfen. *Alle* Zeichen stammen nun aus diesen Schriftarten.

Hier werden nur die Unterschiede zwischen schule2eLua gegenüber schule2e beschrieben. Umdefinitionen und Veränderungen sind im Abschnitt »Das Paket schule2e« schon aufgeführt. Daher hier nur eine Zusammenfassung. Bei Konvertierung von \LaTeX auf lua \LaTeX sollte man die **in dieser Farbe** markierten Befehle ggf. ersetzen oder ändern.

Optionen

Da Schrifteinstellungen wegfallen, ebenso wie Codierungs-Festlegungen, verbleiben nur noch folgende Optionen aus Tab. [A.1 auf der nächsten Seite](#).

Beachte: Alle **pstricks-Pakete** müssen in `\ifluatex\else\usepackage{pstricks, usw.}\fi` eingeschlossen werden.

Auch die Binderandlänge für `typeareaB` kann noch übergeben werden.

Geladene Pakete

Das sind bis auf wenige dieselben wie in schule2e. Allerdings kommen ein paar Lua-spezifische hinzu:

Falls man etwa die astronomischen Symbole von Wasy will, mache man ein `\AtBeginDocument{\usepackage[nointegrals]{wasysym}}` wasy Integrale braucht man ja nicht mehr. Ich habe festgestellt, dass man `\AtBeginDocument` offenbar braucht. **Auch bei anderen Font-change Befehlen wie Umdefinition von `\epsilon` war dies nötig.**

Geänderte, neue oder weggefallene Befehle

Dies betrifft insbesondere Befehle, die neue Schriftarten laden:

- `\dunh` ist weggefallen.
- `\G` und `\U` mussten, weil intern, zu `\GZ` (Grad-Zeichen) und `\Ut` (Unit) werden.
- Die **Integrale** werden default mit `\nolimits` geschrieben. Will man nicht jedesmal `\limits` schreiben, so kann man die neu definierten Schalter `\limU{\langle bigop \rangle}` und `\limR{...}` zur Umschaltung auf `\limits` und zurück für die jeweiligen Integrale (immer je nur eines!) verwenden. Geht auch für andere Operatoren wie `\Sum` usw.

Fonteinstellungen

Die im Abschnitt [1.1.3.2 auf Seite 11](#) aufgeführten Fontauswahlen wurden leicht geändert:

```
\newcommand*{\Kopfzeilenfont}{\small\itshape\vphantom{fgq}}
\newcommand*{\mathCaptionfont}[1]{\ensuremath{\mathsfup{#1}}}
```

Schriftvarianten im Mathematiksatz

Man beachte das Schema von verschiedenen Math-Schriftstilen bei `unicode-math` (im Text-Modus geht's wie gewohnt, aber TeX Gyre Pagella hat keine slanted Glyphen): Dieses System funktioniert auch mit griechischen Buchstaben!

In pdf \LaTeX (mit dem dieses Dokument gesetzt wurde) sind nicht alle Arten (wegen max. 16 Math-Fonts) darstellbar. **Bei Lua und unicode-math gibt es diese Beschränkung nicht mehr.** Siehe auch die Datei `schule2e-Kurz.pdf`, die mit lua \LaTeX übersetzt wurde und alle Varianten anzeigt. Weitere Math-Alphabet-Befehle sind eingebaut:

```
\mathbb, \mathcal, \mathbfcal, \mathscr, \mathbfscr, \mathfrak,
\mathbffrak
```

Eine Liste aller mathematischen Zeichen gibt's als [unimath-symbols.pdf](#).

TAB. A.1: Optionen von `schule2eLua`

<code>brief</code>	<code>sectNA</code>	<code>noGLNrDot</code>	<code>noSectsty</code>	<code>AutoPST</code>	<code>AutoOff</code>	<code>helv</code>
<code>withChapter</code>	<code>tradToc</code>	<code>fnTrenner</code>	<code>pNflush</code>	<code>Pfeilvek</code>	<code>Gothicvek</code>	

TAB. A.2: Von *schule2eLua* geladene Pakete

<code>ifluatex</code>	<code>babel</code>	<code>fontspec</code>	<code>amsmath</code>	<code>unicode-math</code>
<code>luatex-math</code>	<code>microtype</code>	<code>icomma</code>	<code>xspace</code>	<code>units</code>
<code>fancybox</code>	<code>sectsty*</code>	<code>marvosym</code>	<code>braket</code>	<code>amsthm</code>
<code>typeareaB</code>	<code>array</code>	<code>rotating</code>	<code>numprint</code>	<code>tabularx</code>
<code>booktabs</code>	<code>auto-pst-pdf*</code>	<code>url</code>	<code>parskip</code>	<code>etoolbox</code>
<code>luacode</code>	<code>xkeyval</code>			

* Nur mit passender Option

TAB. A.3: `unicode-math` Schriftarten

Befehl	erzeugt	Bemerkung	Variante
—	$a^2 = b^2 + c^2$	<i>roman italic</i>	<code>mathit</code>
<code>mathup</code>	$a^2 = b^2 + c^2$	roman upright	<code>mathrm</code>
<code>mathbf (=bm)</code>	$a^2 = b^2 + c^2$	<i>bold roman italic</i>	<code>mathbf</code>
<code>mathbfup</code>	$a^2 = b^2 + c^2$	bold roman upright	—
<code>mathsfup</code>	—	sans serif aufrecht	—
<code>mathsfit</code>	—	<i>sans serif italic</i>	<code>mathsf</code>
<code>mathbfsfup</code>	—	bold sans serif aufrecht	—
<code>mathbfsfit</code>	—	<i>bold sans serif italic</i>	<code>mathbfsf</code>

B. Obsoletes Umwandeln

Da einige Befehle im Laufe der Zeit weggefallen sind bzw. umbenannt wurden, habe ich eine Emacs Funktion in meiner .emacs, mittels der die Anpassungen automatisch durchgeführt werden können werden können:

```
(defun brf-schule2e-abk ()
  "Ersetzt alle schule2e Abkürzungs-Makros durch die neuen Definitionen."
  (interactive)
  (goto-char (point-min))
  (replace-string "\\zB" "\\abk{zB}")
  (goto-char (point-min))
  (replace-string "\\dH" "\\abk{dh}")
  (goto-char (point-min))
  (replace-string "\\iA" "\\abk{ia}")
  (goto-char (point-min))
  (replace-string "\\uA" "\\abk{ua}")
  (goto-char (point-min))
  (replace-string "\\Pvec" "\\Pvek")
  (goto-char (point-min))
  (replace-string "\\SpVec" "\\Spvek")
  (goto-char (point-min))
  (replace-string "\\MZtoday" "\\today")
  (goto-char (point-min))
  (replace-string "\\mathsbb" "\\mathbb")
  (goto-char (point-min))
  (replace-string "\\textsbb" "\\textbb")
  (goto-char (point-min))
  (replace-string "\\mb " "\\mid ")
  (goto-char (point-min))
  (replace-string "\\mb{" "\\mid{")
  (message "Prüfen: \\np statt \\rc. Für Lua: \\str, \\Str, \\Pvek auf Hoch/Tief")
  (goto-char (point-min))
)
```

Index

A

Ableitungssymbole • 26
abstract • 9
Anmerkungen
 Bibliographie • 13
 hyperref • 15
 Index • 15
 Math-Alphabete • 23
 numprint • 30
 refTeX • 16
 theorem • 14

B

Booklet • 38
bordermatrix • *siehe* Pake-
 te/blkarray
Briefe
 \BitteWenden • 36
 Brief-Klasse • 34
 Schlüsselwörter • 35

C

Captionfont • 12
captions • 29

D

Dezimalkomma • 12
Dvi2Pdf • 39
DviDuplex • 38

E

e und i • 26
Einheitensatz • 27
enumerate • 11

F

Fonteinstellungen • 11
Fraktur und Initiale • 18
 fraktur • 18
 \frakturOn • 18
 \Initial • 18
 \InitialScale • 18
 \textfraktur • 18
 \textInitial • 18
Fußnoten • 17
 \fn • 17
 \fnmark • 18

G

Gleitumgebungen • 11
GMS-Eingabe • *siehe* HMS-
 Eingabe
Gradzeichen • 25
Griechische Buchstaben • 8,
 13, 24

H

headings • *siehe* Kopfzeilen
Hilfsprogramme • 38
HMS-Eingabe • 25
 \HMS, \GMS • 25
 \HMSDez, \HMSFig • 25

I

Inhaltsverzeichnis • 6

K

Klassen
 KOMA • 21, 33

revTeX4-1 • 13

Kopfzeilen • 20

L

Längen • 16
 \breite, \Breite • 16
 \TALength • 16
leqno • 27

M

Makros

 \abk • 25
 \areaset • 33
 \ArrF • 29
 \bigcupdot • 28
 \BildTop • 31
 \bm • 25
 \braket, \bra, \ket • 26
 \BS • 24
 \caps • 23, 24
 \Cdot • 26
 \clearmpydoublepage • 20
 \CMcal • 8
 \cupdot • 28
 \D • 26
 \dD • 26
 \diff • 26
 \dunh • 22
 \E, \I • 26
 \EnglischeLinie • 25
 \entspricht • 26
 \epsilon • 24
 \euro mit Varianten • 24
 \evек • 28

`\Fixdate` • 29
 Fonts
 `\Abschnittfont` • 12
 `\Captionfont` • 12
 `\Kopfzeilenfont` • 12
 `\Labelingfont` • 12
 `\mathCaptionfont` • 12
 `\Paragraphfont` • 12
 `\textAufgabenfont` • 12
 `\textCapTitFont` • 12
 `\textMinisecfont` • 12
`\G, \GC, \GF, \GZ` • 25
`\geq` • 24
`\Hang` • 22
`\iD` • 26
`\im` • 26
`\IM` • 28
`\iniAufg` • 20
`\inlineintertext` • *siehe*
 `\vorintertext`
`\keinPlatz` • 31
`\Kopfzeile` • *siehe* Kopfzeilen
`\kurzintertext` • 27
`\leq` • 24
`\limR` • 43
`\limU` • 43
`\lokalhref` • 16
`\mathbb` • 22
`\mathfig` • 41
`\mathfrak` • 23
`\mathscr` • 23
`\mathsfb, \textsfb` • 23
`\mcaps` • 24
`\mdiff` • 26
`\Menge` • 26
`\minisec` • 21
`\MRightarrow` • 25
`\MRzell` • 31
`\todayInter` • 29
`\NA, \NL` • 19
`\nfrac` • 26
`\npboldtext` • 30
`\oiint` • 28
`\Op` • 28
`\parindentOff` • 5, 12
`\pD` • 26
`\pdiff` • 26
`\Pkt` • 26
`\printindex` • 15
`\psfragfig` • 41
`\Punkte, \mPunkte` • 19
`\Pvek` • 27
`\qbox, \qqbox` • 29
`\raggedhyphen` • 22
`\re` • 26
`\Set` • 26
`\Setzbreiten` • 29
`\SFZ` • 25
`\shortintertext` • *siehe*
 `\kurzintertext`
`\Spvek` • 28
`\sqrt` • 24
`\Sr` • 29
`\starlabel` • 27
`\sternchen` • 25
`\str` • 27
`\Str` • 27
`\Sumbreite` • 29
`\T` • 26
`\TA, \TAspace` • 20
`\TabBild` • 31
`\TabV` • 29
`\textbb` • 22
`\Titelzeile` • 22
`\TO` • 27
`\tPvek` • 27
`\tstr` • 27
`\tStr` • 27
`\TU` • 27
`\typearea` • 33
`\U` • 27
`\Uhr` • 29
`\URL` • 9, 16
`\vek` • 28
`\Versal` • 24
`\vH` • 25
`\vorintertext` • 27
`\Winkel` • 28
`\Wurzel` • 24
`\Z` • 28
`\zeilstyle` • 31
 Maßeinheiten • 26
O
 Operator-Symbole • 26
 Optionen • 5
 alle • 5
 ansi • 6
 AutoOff • 8
 AutoPST • 8
 Avant • 7
 Binderand • 7
 brief • 5
 Eucal • 8
 fnTrenner • 6
 Gothicvek • 8, 28
 helv • 7
 latin1 • 6
 mac • 6
 noEco • 7
 noGLNrDot • 6
 noLinespread • 6
 noSectsty • 6
 ora • 9
 orgGreek • 8
 palatino • 7
 Pfeilvek • 8, 28
 pNflush • 7
 pxBBold • 8
 sectNA • 7
 SkriptFont • 8
 tradToc • 6
 utf8 • 6
 WasyFont • 8
 WasyInt • 8
 withChapter • 6
 optischer Randausgleich • 9

P

pagestyle • *siehe* Kopfzeilen

Pakete • 9

auto-pst-pdf • 8, 40

babel • 5, 9

blkarray • 31

circ • 9

hyperref • 15

marvosym • 9

microtype • 24

ngerman • 9

numprint • 30

palateuler • 37

reftex • 16

sectsty • 13

tensind • 36

tensindbrf • 36

typeareaB • 33

units • 27

url • 9, 15

wasysym • 9, 43

Pakete, eingebundene • 9

Papierformate • 34

Pfeile • 26

S

Seitennummern • 7

Serienbriefe • 36

Spaltenvektoren • 28

Sperrung von Text • 23

T

Tabellen • 29

Tabellen-Spaltenstile • 30

Tensoren • 26, 36

U

Umgebungen

CBeschreibung • 21

hangingpars • 22

labeling • 21

merkbox • 21

oldstyletabellen • 29

TAufg, TTAufg • 20

TTAufgVar • 20

Ungleichheitszeichen • 24

W

webo-Fonts • 40

Z

Zähler • 16

aufgabennr, teilaufgabennr • 16

Zehnerpotenzen • 28