

**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**

---

<https://tu-dresden.de/cd>

---

TUD-Script v2.06o basierend auf KOMA-Script

# Ein $\text{\LaTeX}$ -Bundle für Dokumente im Corporate Design der Technischen Universität Dresden

Benutzerhandbuch\*

Falk Hanisch

[hanisch.latex@outlook.com](mailto:hanisch.latex@outlook.com)

9. August 2022

<https://github.com/tud-cd/tudscr>

\*Online-Version

# Vorwort

In diesem Handbuch werden die für das Erstellen von  $\LaTeX$ -Dokumenten im Corporate Design der Technischen Universität Dresden<sup>1</sup> entwickelten Klassen und Pakete beschrieben. Diese basieren auf den gerade im deutschsprachigen Raum häufig verwendeten KOMA-Script-Klassen, die eine Vielzahl von Einstellmöglichkeiten bieten, welche weit über die Möglichkeiten der  $\LaTeX$ -Standardklassen hinausgehen. Zusätzlich bietet das TUD-Script-Bundle weitere, insbesondere das Dokumentlayout betreffende Auswahlmöglichkeiten.

Es sei angemerkt, dass die hier beschriebenen Klassen eine Abweichung vom Corporate Design der Technischen Universität Dresden zulassen, da dieses gerade unter typografischen Gesichtspunkten durchaus als diskussionswürdig zu erachten ist. Prinzipiell ist es mit den entsprechenden Einstellungen möglich, auf das Layout der KOMA-Script-Klassen zurückzuschalten. Ohne die gezielte Verwendung dieser Optionen durch den Anwender werden per Voreinstellung alle Vorgaben des Corporate Designs umgesetzt.

Dieses Handbuch enthält eine Beschreibung der *zusätzlich* zu den KOMA-Script-Klassen nutzbaren Optionen und Befehle. Dabei werden Grundkenntnisse in der Verwendung von  $\LaTeX$  vorausgesetzt. Sollten diese nicht vorhanden sein, wird das Lesen der  $\LaTeX$  2 $\epsilon$ -Kurzbeschreibung **l2kurz.pdf**<sup>2</sup> dringend empfohlen. Für den vertiefenden Einstieg in  $\LaTeX$  stellt Nicola L. C. Talbot sehr ausführliche Tutorials für  $\LaTeX$ -Novizen<sup>3</sup> und Dissertationen<sup>4</sup> zur freien Verfügung. Zusätzlich werden in Teil II dieses Handbuchs Minimalbeispiele sowie etwas ausführlichere Tutorials angeboten.

Des Weiteren sollte *jeder* Anwender das  $\LaTeX$  2 $\epsilon$ -Sündenregister **l2tabu.pdf**<sup>5</sup> kennen, um typische Fehler zu vermeiden. Antworten auf häufig gestellte Fragen liefert DanteFAQ<sup>6</sup>. Falls der Nutzer unerfahren bei der Verwendung von KOMA-Script sein sollte, so ist ein Blick in das dazugehörige Handbuch<sup>7</sup> sehr zu empfehlen, wenn nicht sogar unumgänglich.

Der aktuelle Stand der Klassen und Pakete aus dem TUD-Script-Bundle in der Version v2.06o wurde nach bestem Wissen und Gewissen auf Herz und Nieren getestet. Dennoch kann nicht für das Ausbleiben von Fehlern garantiert werden. Beim Auftreten eines Problems sollte dieses genauso wie Inkompatibilitäten mit anderen Paketen im TUD- $\LaTeX$ -Forum respektive im GitHub-Repository **tud-cd/tudscr** unter

<http://latex.wcms-file3.tu-dresden.de/phpBB3/>

<https://github.com/tud-cd/tudscr/issues>

gemeldet werden. Für eine schnelle und erfolgreiche Fehlersuche sollte bitte ein **Minimalbeispiel**<sup>8</sup> bereitgestellt werden. Auf Anfragen ohne dieses werde ich gegebenenfalls verspätet oder gar nicht reagieren. Ebenso sind im genannten Forum auch *Fragen, Kritik* und *Verbesserungsvorschläge* – sowohl das Bundle selbst als auch die Dokumentation betreffend – gerne gesehen. Da dieses Bundle in meiner Freizeit entstanden ist und auch gepflegt wird, bitte ich um Nachsicht, falls ich nicht sofort antworte und/oder eine Fehlerkorrektur vornehmen kann.

Falk Hanisch

Dresden, 9. August 2022

---

<sup>1</sup><https://tu-dresden.de/cd/>

<sup>2</sup><http://mirrors.ctan.org/info/lshort/german/l2kurz.pdf>

<sup>3</sup><http://www.dickimaw-books.com/latex/novices/>

<sup>4</sup><http://www.dickimaw-books.com/latex/thesis/>

<sup>5</sup><http://mirrors.ctan.org/info/l2tabu/german/l2tabu.pdf>

<sup>6</sup><http://projekte.dante.de/DanteFAQ/WebHome>

<sup>7</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

<sup>8</sup><http://www.komascript.de/minimalbeispiel>

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>9</b>
1.1	Bestandteile und Nutzung des TUD-Script-Bundles . . . . .	10
1.2	Zur Verwendung dieses Handbuchs . . . . .	11
1.3	Schnelleinstieg . . . . .	11
<b>I</b>	<b>Das TUD-Script-Bundle</b>	<b>13</b>
<b>2</b>	<b>Die Hauptklassen tudscrbook, tudscrreprt und tudscrtcl</b>	<b>14</b>
2.1	Die Schriften des Corporate Designs . . . . .	15
2.1.1	Schriften für den Textsatz . . . . .	16
2.1.2	Schriften für den Mathematiksatz . . . . .	19
2.1.3	Vertikaler Leerraum in Abhängigkeit der Schriftgröße . . . . .	21
2.2	Das Layout des Corporate Designs . . . . .	22
2.2.1	Der Satzspiegel . . . . .	22
2.2.2	Die Gestalt von Titel, Umschlagseite, Teilen sowie Kapiteln & Co. . . . .	24
2.2.3	Seiten im Stil des Corporate Designs . . . . .	26
2.2.4	Der Titel und die Umschlagseite . . . . .	32
2.2.5	Die Teileseite . . . . .	39
2.2.6	Die Kapitelseite . . . . .	40
2.2.7	Vakatseiten . . . . .	41
2.2.8	Verwendung von Schriftelementen . . . . .	42
2.2.9	Verwendung von Feldinhalten . . . . .	42
2.2.10	Die Farben des Corporate Designs . . . . .	42
2.3	Zusätzliche Optionen und Erweiterungen . . . . .	42
2.3.1	Zusammenfassung/Kurzfassung . . . . .	42
2.3.2	Selbstständigkeitserklärung und Sperrvermerk . . . . .	46
2.3.3	Lesezeichen . . . . .	50
2.4	Sprachabhängige Bezeichner . . . . .	50
2.5	Kompatibilitätseinstellungen zu früheren Versionen . . . . .	55
<b>3</b>	<b>Die Posterklasse tudscrposter</b>	<b>57</b>
3.1	Layout und Formatierung eines Posters . . . . .	58
3.1.1	Die Wahl von Papierformat und Schriftgröße . . . . .	59
3.1.2	Die Gestalt eines Posters . . . . .	60
3.2	Felder für den Fußbereich . . . . .	61
3.3	Sprachabhängige Bezeichner für den Fußbereich . . . . .	62
<b>4</b>	<b>Das Paket tudscrsupervisor – Studentische Betreuung</b>	<b>63</b>
4.1	Aufgabenstellung für eine wissenschaftliche Arbeit . . . . .	64
4.2	Gutachten für wissenschaftliche Arbeiten . . . . .	65
4.3	Aushang . . . . .	66
4.4	Zusätzliche sprachabhängige Bezeichner . . . . .	67
<b>5</b>	<b>Zusätzliche Pakete im TUD-Script-Bundle</b>	<b>70</b>
5.1	Das Paket tudscrcolor – Farben im Corporate Design . . . . .	71
5.1.1	Generelle Farbdefinitionen . . . . .	71
5.1.2	Zusätzliche Farbdefinitionen . . . . .	73

5.1.3	Umstellung des Farbmodells . . . . .	73
5.2	Das Paket <b>tudscrfonts</b> – Schriften im Corporate Design . . . . .	73
5.3	Das Paket <b>fix-tudscrfonts</b> – Schriftkompatibilität . . . . .	74
5.4	Das Paket <b>mathswap</b> . . . . .	75
5.5	Das Paket <b>twocolfix</b> . . . . .	75
<b>II</b>	<b>Ergänzungen und Hinweise</b>	<b>77</b>
<b>6</b>	<b>Minimalbeispiele</b>	<b>78</b>
6.1	Dokument . . . . .	79
6.2	Dissertation . . . . .	84
6.3	Abschlussarbeit (kollaborativ) . . . . .	87
6.4	Aufgabenstellung (kollaborativ) . . . . .	89
6.5	Gutachten . . . . .	91
6.6	Aushang . . . . .	93
6.7	Poster . . . . .	95
<b>7</b>	<b>Tutorials</b>	<b>97</b>
<b>8</b>	<b>Benötigte, unterstützte und empfehlenswerte Pakete</b>	<b>99</b>
8.1	Notwendige und ergänzende Pakete für TUD-Script . . . . .	100
8.2	Empfehlenswerte Pakete . . . . .	101
8.2.1	Pakete zur Verwendung in jedem Dokument . . . . .	101
8.2.2	Pakete zur situativen Verwendung . . . . .	102
<b>9</b>	<b>Praktische Tipps &amp; Tricks</b>	<b>115</b>
9.1	$\LaTeX$ -Editoren . . . . .	116
9.2	Literaturverwaltung in $\LaTeX$ . . . . .	116
9.3	Worttrennungen in deutschsprachigen Texten . . . . .	117
9.4	Lokale Änderungen von Befehlen und Einstellungen . . . . .	118
9.5	Bezeichnung der Gliederungsebenen durch <b>hyperref</b> . . . . .	118
9.6	URL-Umbrüche im Literaturverzeichnis mit <b>biblatex</b> . . . . .	119
9.7	Zeilenabstände in Überschriften . . . . .	119
9.8	Warnung wegen zu geringer Höhe der Kopf-/Fußzeile . . . . .	119
9.9	Einrückung von Tabellenspalten verhindern . . . . .	119
9.10	Unterdrückung des Einzuges eines Absatzes . . . . .	120
9.11	Unterbinden des Zurücksetzens von Fußnoten . . . . .	120
9.12	Warnung beim Erzeugen des Inhaltsverzeichnisses . . . . .	120
9.13	Leer- und Satzzeichen nach $\LaTeX$ -Befehlen . . . . .	121
9.14	Finden von unbekanntem $\LaTeX$ -Symbolen . . . . .	121
9.15	Das Setzen von Auslassungspunkten . . . . .	121
9.16	Lokalisierung für das Setzen von Einheiten mit <b>siunitx</b> . . . . .	122
9.17	Fehlermeldung beim Laden eines Paketes mit Optionen . . . . .	122
9.18	Probleme bei der Verwendung von <b>auto-pst-pdf</b> . . . . .	122
9.19	Automatisiertes Einbinden von <b>Inkscape</b> -Grafiken . . . . .	123
9.20	Änderung des Papierformates . . . . .	123
9.21	Beschnittzugabe und Schnittmarken . . . . .	124

9.22 Vermeiden des Skalierens einer PDF-Datei beim Druck . . . . .	125
9.23 Platzierung von Gleitobjekten . . . . .	125
9.24 Warnung bei der Schriftgrößenwahl . . . . .	126
9.25 Fehlermeldung: ! No room for a new \write . . . . .	127
<b>III Anhang</b>	<b>128</b>
<b>A Weiterführende Installationshinweise</b>	<b>129</b>
A.1 Lokale Deinstallation des TUD-Script-Bundles . . . . .	131
A.2 Lokale Installation des TUD-Script-Bundles . . . . .	132
A.3 Lokales Update des TUD-Script-Bundles . . . . .	132
A.3.1 Update des TUD-Script-Bundles ab Version v2.02 . . . . .	132
A.3.2 Update des TUD-Script-Bundles ab Version v2.00 . . . . .	132
A.3.3 Update des TUD-Script-Bundles von Version v1.0 . . . . .	132
A.4 Installation veralteter Schriftfamilien . . . . .	133
A.4.1 Installation der Type1-Schriften unter Windows . . . . .	135
A.4.2 Installation der Type1-Schriften unter Linux und OS X . . . . .	135
A.4.3 Installationshinweise für portable Installationen . . . . .	136
A.4.4 Probleme bei der Installation der Type1-Schriften . . . . .	137
<b>B Obsolete sowie vollständig entfernte Optionen und Befehle</b>	<b>138</b>
B.1 Veraltete Optionen und Befehle in TUD-Script . . . . .	139
B.1.1 Änderungen für TUD-Script v2.00 . . . . .	139
B.1.2 Änderungen für TUD-Script v2.02 . . . . .	141
B.1.3 Änderungen für TUD-Script v2.03 . . . . .	142
B.1.4 Änderungen für TUD-Script v2.04 . . . . .	142
B.1.5 Änderungen für TUD-Script v2.05 . . . . .	142
B.1.6 Änderungen für TUD-Script v2.06 . . . . .	143
B.2 Das Paket <b>tudscrcomp</b> – Umstieg von anderen Klassen . . . . .	144
B.2.1 Optionen und Befehle aus <b>tudbook &amp; Co.</b> . . . . .	145
B.2.2 Optionen und Befehle aus <b>tudmathposter</b> . . . . .	146
<b>C Identifikation von TUD-Script</b>	<b>149</b>
<b>Index</b>	<b>151</b>
Allgemeiner Index . . . . .	152
Klassen- und Paketooptionen . . . . .	155
Befehle und Umgebungen mit zugehörigen Parametern . . . . .	157
Sprachabhängige Bezeichner . . . . .	160
Seitenstile, Schriftelemente und Farben . . . . .	161
Längen und Zähler . . . . .	162
Klassen, Pakete und Dateien . . . . .	162
<b>Änderungsliste</b>	<b>165</b>
v2.00 . . . . .	166
v2.01 . . . . .	167

v2.02	167
v2.03	169
v2.04	171
v2.05	172
v2.06	173



**Kapitel 1**

**Einleitung**

Zur Verwendung von TUD-Script v2.06o wird sowohl KOMA-Script v3.36 oder später als auch die Hausschrift des Corporate Designs **Open Sans** aus dem Paket **opensans** zwingend benötigt. Außerdem müssen durch die genutzte  $\LaTeX$ -Distribution weitere Pakete bereitgestellt werden, die unter Unterkapitel 8.1 aufgeführt sind. Beim Einsatz einer der Distributionen **T $\TeX$  Live**, **Mac $\TeX$**  und **MiK $\TeX$**  in der jeweils aktuellen Versionen sollte dies kein Problem darstellen. Wird jedoch eine Distribution verwendet, die TUD-Script in der Version v2.06o nicht zur Verfügung stellt und eine Aktualisierung dieser nicht möglich sein, so sollte Anhang A konsultiert werden.

Das TUD-Script-Bundle ist hauptsächlich für das Erstellen wissenschaftlicher Texte sowie Arbeiten gedacht und soll *momentan* die ursprünglichen Klassen<sup>9</sup> aus dem Vorlagenpaket von Klaus Bergmann nicht ersetzen sondern vielmehr ergänzen. Eine Umsetzung des Corporate Designs für die **beamer**-Klasse sowie für Briefe und Geschäftsschreiben auf Basis des KOMA-Script-Brief-Klasse **scrletter** ist bis jetzt leider noch nicht mit TUD-Script realisiert worden, soll jedoch langfristig erfolgen. Im Bundle **tudmathposter** existieren für die **beamer**-Klasse allerdings bereits mehrere Stile. Dieses ist im GitHub-Repository **tud-cd**<sup>10</sup> zu finden. Für das Erstellen von Briefen mit den TUD-Script-Klassen ließe sich das Paket **scrletter** nutzen.

## 1.1 Bestandteile und Nutzung des TUD-Script-Bundles

v2.01  
v2.02  
v2.04  
v2.06

Das TUD-Script-Bundle wird über das Comprehensive TeX Archive Network (CTAN)<sup>11</sup> bereitgestellt und kann durch  $\LaTeX$ -Distributionen wie **T $\TeX$  Live**, **Mac $\TeX$**  oder auch **MiK $\TeX$**  genutzt werden. Es stellt zum einen die drei Hauptklassen **tudscrbook**, **tudscrrcpt** sowie **tudscartcl** und zum anderen die Klasse **tudscrposter** zur Verfügung, welche in Kapitel 2 beziehungsweise Kapitel 3 vorgestellt werden. Das Paket **tudscrsupervisor** – dokumentiert in Kapitel 4 – kann zusammen mit den TUD-Script-Klassen genutzt werden, um Aufgabenstellungen, Aushänge oder Gutachten für studentische Arbeiten zu erstellen. Weiterhin existieren auch eigenständige Pakete, welche in Kapitel 5 beschrieben sind. Für die Verwendung des TUD-Script-Bundles ist – neben KOMA-Script mindestens in der Version v3.36 sowie den in Unterkapitel 8.1 aufgeführten  $\LaTeX$ -Paketen – seit der Version v2.06 lediglich die Schrift **Open Sans** vonnöten, welche durch das Paket **opensans** zur Verfügung gestellt wird. Eine lokale Nutzerinstallation der Schriften – wie in vorherigen Versionen – ist nicht notwendig. Lediglich für den Fall, dass gezielt die alten Schriften **Univers** und **DIN Bold** eingesetzt werden sollen, müssen diese auch installiert sein. Weitere Hinweise dazu sind in Unterkapitel A.4 zu finden. Um diese zu aktivieren, sei auf `tudscrver=2.05` und `cdo1dfont` verwiesen.

### Anmerkung zu Windows

Für Windows können zwei unterschiedliche  $\LaTeX$ -Distributionen genutzt werden. Die Vorteile von **T $\TeX$  Live** im Vergleich zu **MiK $\TeX$**  liegen zum einen in der Wartung durch mehrere Autoren sowie der früheren Verfügbarkeit aller Updates über CTAN. Zum anderen werden zusätzlich zu  $\LaTeX 2_{\epsilon}$  ein *Perl-Interpreter* sowie *Ghostscript* mitgeliefert, wodurch die Ad-hoc-Verwendung einiger Pakete wie beispielsweise **glossaries** vereinfacht beziehungsweise verbessert wird. Für **MiK $\TeX$**  müssen diese Programme gegebenenfalls manuell installiert werden. Demgegenüber entfällt für **MiK $\TeX$**  die alljährliche Neuinstallation, welche bei **T $\TeX$  Live** notwendig ist.

<sup>9</sup>**tudbook**, **tudbeamer**, **tudletter**, **tudfax**, **tudhaus**, **tudform**

<sup>10</sup><https://github.com/tud-cd/tud-cd/>

<sup>11</sup><http://www.ctan.org/>

## Anmerkung zu Linux und OS X

Die Installation einer der  $\text{\LaTeX}$ -Distributionen **T $\text{\E}$ X Live** oder **MacT $\text{\E}$ X** sollte direkt über die angebotenen Pakete (<https://tug.org/texlive/> oder <https://tug.org/mactex/>) und nicht über `apt-get install` erfolgen. Damit wird sichergestellt, dass die aktuelle Variante der jeweiligen Distribution genutzt wird.

## 1.2 Zur Verwendung dieses Handbuchs

Sämtliche neu definierten Optionen, Umgebungen und Befehle der Klassen und Pakete des TUD-Script-Bundles werden im Handbuch aufgeführt und beschrieben. Am Ende des Dokumentes befinden sich mehrere Indexe, die das Nachschlagen oder Auffinden von bisher unbekanntem Befehlen oder Optionen erleichtern sollen.

Die im Folgenden beschriebenen Optionen können – wie ein Großteil aller Einstellungen für KOMA-Script – in der Syntax des **keyval**-Paketes als Schlüssel-Wert-Paare bei der Wahl der Dokumentklasse angegeben werden: `\documentclass[<Schlüssel>=<Wert>]{<Klasse>}`

Des Weiteren eröffnen die KOMA-Script-Klassen die Möglichkeit der späten Optionenauswahl. Damit können Optionen nicht nur direkt beim Laden als sogenannte Klassenoptionen angegeben werden, sondern lassen sich auch noch innerhalb des Dokumentes nach dem Laden der Klasse ändern. Die KOMA-Script-Klassen sehen hierfür zwei Befehle vor. Mit `\KOMAOPTIONS{<Optionenliste>}` lassen sich beliebig viele Schlüssel jeweils genau einen Wert zuweisen, `\KOMAOPTION{<Option>}{<Werteliste>}` erlaubt das gleichzeitige Setzen mehrerer Werte für genau einen Schlüssel. Für die von TUD-Script *zusätzlich* zur Verfügung gestellten Optionen werden äquivalent dazu die Befehle `\TUDOPTIONS{<Optionenliste>}` und `\TUDOPTION{<Option>}{<Werteliste>}` definiert. Damit kann das Verhalten von Optionen im Dokument – innerhalb einer Gruppe auch lokal – geändert werden.

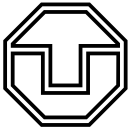
Die Voreinstellung jeder Option wird mit „Standardwert: <Wert>“ bei deren Beschreibung angeführt. Einige dieser Voreinstellungen sind nicht immer gleich sondern werden in Abhängigkeit der genutzten Benutzereinstellungen und Optionen gesetzt. Diese bedingten Voreinstellungen werden durch „Standardwert: <Wert> | Bedingung: <bedingter Wert>“ angegeben. Wird ein Schlüssel durch den Benutzer *ohne* eine Wertzuweisung genutzt, so wird – falls vorhanden – ein vordefinierter Säumniswert gesetzt, welcher in der Beschreibung aller Optionen durch die *kursive* Schreibweise innerhalb der Werteliste gekennzeichnet ist. In den meisten Fällen ist der Säumniswert eines Schlüssels `true`, er entspricht folglich der Angabe `<Schlüssel>=true`. Mit der expliziten Wertzuweisung eines Schlüssels durch den Benutzer werden sowohl einfache als auch bedingte Voreinstellungen immer überschrieben. Die neben den Optionen neu eingeführten Befehle und Umgebungen der Klassen werden im gleichen Stil erläutert.

## 1.3 Schnelleinstieg

Das Handbuch gliedert sich in drei Teile. In Teil I ist die Dokumentation von TUD-Script zu finden. Hier werden alle neuen Optionen, Umgebungen und Befehle, die über die Funktionalität von KOMA-Script hinausgehen, erläutert. Teil II enthält zum einen einfache Minimalbeispiele, um den prinzipiellen Umgang und die Funktionalitäten von TUD-Script zu demonstrieren. Zum anderen werden hier auch ausführliche und dokumentierte Tutorials vor allem für  $\text{\LaTeX}$ -Neulinge angeboten. Insbesondere das Tutorial **treatise.pdf** ist mehr als einen Blick wert, wenn eine wissenschaftliche Arbeit mit  $\text{\LaTeX}$  2 $\epsilon$  verfasst werden soll. Abschließend werden verschiedenste

Pakete vorgestellt, die nicht speziell für das TUD-Script-Bundle selber sondern auch für andere  $\LaTeX 2_{\epsilon}$ -Klassen verwendet werden können und demzufolge für jeden Anwender interessant sein könnten. Außerdem werden hier einige Tipps & Tricks beim Umgang mit  $\LaTeX$  beschrieben, um kleinere oder größere Probleme zu lösen.

Die Klassen **tudscrbook**, **tudscrreprt** und **tudscartcl** sind Wrapper-Klassen der bekannten KOMA-Script-Klassen **scrbook**, **scrreprt** sowie **scartcl** und können einfach anstelle deren verwendet werden. Auf diesen basierende Dokumente können durch das Umstellen der Dokumentklasse einfach in das Corporate Design der Technischen Universität Dresden überführt werden. Bei Fragestellungen bezüglich Layout, Schriften oder ähnlichem ist in jedem Fall ein weiterer Blick in das hier vorliegende Handbuch empfehlenswert.



## Teil I

# Das TUD-Script-Bundle

Dies ist der Hauptteil des TUD-Script-Bundles. Hier findet der Anwender alle verfügbaren Optionen, Umgebungen und Befehle, die über die Funktionalität von KOMA-Script hinausgehen.

## Kapitel 2

# Die Hauptklassen

Es werden die drei neuen Hauptklassen

**tudscrbook**  
**tudscrreprt**  
**tudscrtartcl**

eingeführt, welche auf den KOMA-Script-Klassen basieren und grundsätzlich alle deren bereitgestellte Optionen – beispielsweise `parskip=<Methode>` für die Absatz Einstellungen oder `BCOR=<Längenwert>` zur Festlegung der Bindekorrektur – sowie Umgebungen und Befehle unterstützen. Zusätzlich zu den KOMA-Script-Klassen werden weitere Pakete zwingend benötigt, welche unter Unterkapitel 8.1 aufgeführt sind und durch TUD-Script geladen werden.

Es sei hier abermals auf die Anwenderdokumentation von KOMA-Script<sup>12</sup> hingewiesen, viele der folgend beschriebenen Befehle und Optionen beziehen sich auf die darin vorgestellten Einstellungsmöglichkeiten. Die Anpassungen und Erweiterungen der KOMA-Script-Klassen an das Corporate Design und die neu definierten beziehungsweise geänderten oder erweiterten Befehle und Optionen werden im Folgenden erläutert.

```
\TUOptions{<Optionenliste>}  
\TUOption{<Option>}{<Werteliste>}
```

Mit diesen beiden Befehlen besteht für die meisten der neuen Klassen- respektive Paketoptionen die Möglichkeit, den Wert der Optionen noch nach dem Laden der Klasse beziehungsweise des Paketes zu ändern. Sie sind das Äquivalent zu den beiden KOMA-Script-Befehlen `\KOMAoptions` beziehungsweise `\KOMAoption`.

Man kann wahlweise mit der Anweisung `\TUOptions` die Werte mehrerer Optionen gleichzeitig ändern, wobei diese durch Kommata zu trennen sind. Dabei muss innerhalb der Optionenliste jede zu ändernde Option in der Form `<Option>=<Wert>` angegeben werden. Die meisten der TUD-Script-Optionen besitzen auch einen Standard- respektive Säumniswert. Werden diese in der Form `<Option>` ohne die Angabe eines Wertes genutzt, so wird der jeweiligen Option einfach der vorgesehene Säumniswert zugewiesen.

Einige Optionen können gleichzeitig mehrere, sich ergänzende Werte besitzen. Für diese besteht die Möglichkeit, mit `\TUOption` einer einzelnen Option nacheinander eine Reihe von Werten zuzuweisen, wobei diese in der Werteliste durch Komma voneinander zu trennen sind.

Diese beiden Befehlen erlauben es im Bedarfsfall das Verhalten von einer Option oder mehreren Optionen im Dokument zu ändern. Werden diese Befehle in einer Umgebung oder Gruppe genutzt, bleiben die vorgenommenen Einstellungen innerhalb dieser lokal begrenzt.

## 2.1 Die Schriften des Corporate Designs

Das Corporate Design der Technischen Universität Dresden gibt die Verwendung der Schriftfamilie **Open Sans** für den Fließtext vor, was in der Standardkonfiguration durch TUD-Script auch so umgesetzt wird. Da jedoch in längeren Texten die Verwendung einer Serifenschriften aus typografischer Sicht zu empfehlen ist, gibt es allerdings die Möglichkeit, die ursprünglich vorgesehenen Schriften des Corporate Designs nicht zu laden und stattdessen die  $\text{\LaTeX}$ -Standardschriften beziehungsweise ein anderes Schriftpaket zu verwenden. Die Erläuterungen dazu sind in Abschnitt 2.1.1 zu finden. Zur Verwendung der Schriften mit  $\text{\LaTeX}$  2 $\epsilon$ -Klassen, welche nicht zum TUD-Script-Bundle gehören, lässt sich das Paket **tudscrfonts** laden.

<sup>12</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

v2.02  
v2.04  
v2.06

Durch das Corporate Design werden keine Schriften für den Mathematiksatz festgelegt. Das ist insbesondere für sowohl mathematische Abhandlungen als auch ingenieur- und naturwissenschaftliche Dokumente nicht tragbar. Im Mathematikmodus werden deshalb die lateinischen Glyphen mithilfe des Paketes **mathastext** sowie die griechischen Lettern der **Open Sans** genutzt. Zu Ergänzung kann für zusätzliche mathematische Symbole das Paket **mdsymbol** geladen werden.

Diese Einstellung lässt sich deaktivieren, wodurch sich die Standardschriften oder gegebenenfalls die eines zusätzlichen Paketes für den mathematischen Satz nutzen lassen. Die dafür relevanten Einstellungen werden in Abschnitt 2.1.2 erläutert. Weiterhin sind ergänzende Hinweise zu einem typografisch sauberen Mathematiksatz in Kapitel 7 zu finden.

## Die Schriftformate Type1 und OpenType

v2.02

Das TUD-Script-Bundle unterstützt die Nutzung der Schriften des Corporate Designs sowohl im Type1- als auch im OpenType-Format. Beide Formate werden durch das Paket **opensans** bereitgestellt, wobei unabhängig vom genutzten Textsatzsystem die Schriften im Type1-Format geladen werden. Die Verwendung der Schriften im OpenType-Format ist auch beim Einsatz von **Lua $\LaTeX$**  oder **X $\LaTeX$**  nicht zwingend erforderlich. Dennoch ist dies problemlos möglich, indem einfach das Paket **fontspec** eingebunden wird.

Ein Verzicht auf die Type1-Schriften ist dennoch nicht ohne Weiteres möglich. Denn einerseits sind diese für die Kompilierung eines Dokumentes über den klassischen Prozesspfad via `latex > dvips > ps2pdf` – wie es beispielsweise für die Erstellung von Grafiken mit **pstricks** notwendig ist – erforderlich. Andererseits werden für den Mathematiksatz die Glyphen den Type1-Schriften entnommen. Eine Verwendung der OpenType-Schriften für den Mathematiksatz ist leider nicht ohne einen erheblichen Aufwand möglich, weshalb TUD-Script – insbesondere aufgrund des nicht vorhandenen Mehrwerts – darauf verzichtet.

### 2.1.1 Schriften für den Textsatz

v2.02  
v2.03  
v2.05  
v2.06

`cdfont=<Einstellung>` Voreinstellung: `true,normalbold,liningfigures`

Mit der Option `cdfont` können durch den Anwender alle zentralen Schrifteinstellungen für die TUD-Script-Klassen vorgenommen werden. Dies betrifft die Schriften sowohl für den Fließtext als auch den Mathematiksatz. Dabei lassen sich insbesondere unterschiedliche Kombinationen von normaler und fetter Schriftstärke für die **Open Sans** einstellen. Zu Beginn des Dokumentes sind diese maßgeblich für die Definition der Mathematikschriften. Die Schriftstärke im charakteristischen Querbalken der Kopfzeile lässt sich mit dieser Option ebenfalls einstellen.

`cdfont=false/off/no`

Es werden die  $\LaTeX$ -Standardschriften und nicht die Hausschrift des Corporate Designs verwendet. Der Anwender kann beliebige Schriftpakete nutzen.<sup>13</sup>

`cdfont=true/on/yes/light/lightfont/noheavyfont` (Säumniswert)

Es wird die Hausschrift **Open Sans** im Stil des Corporate Designs genutzt. Die Überschriften der obersten Gliederungsebenen bis einschließlich `\subsubsection` werden entweder in fetter oder extra-fetter Schriftstärke gesetzt (*siehe* `headings=<Einstellung>` auf Seite 25), darunter liegende<sup>14</sup> verwenden immer den fetten Schriftschnitt. Dieser nutzt standardmäßig **Open Sans Regular**, kann allerdings mit `cdfont=heavybold` stärker eingestellt werden. Im Fließtext kommt Open Sans Light zum Einsatz.

<sup>13</sup>Für die Verwendung der klassischen  $\LaTeX$ -Schriften, ist das Paket **lmodern** sehr empfehlenswert.

<sup>14</sup>`\paragraph` und `\subparagraph`



`cdfont=heavy/heavyfont`

Diese Einstellung unterscheidet sich von `cdfont=true` insoweit als die Schriftstärke der Hausschrift erhöht wird. Der Fließtext des Dokumentes wird in **Open Sans Regular** gesetzt. Der fette Schriftschnitt ist im Normalfall auf **Open Sans Semi-Bold** festgelegt.

Die Stärke der fetten Schriften lässt sich mit folgenden Einstellungen anpassen.

v2.05

`cdfont=normalbold`

Für die fette Schriftstärke wird **Open Sans Regular** respektive bei stärkerer Grundschrift (`cdfont=heavy`) **Open Sans Semi-Bold** verwendet. Dies ist die Voreinstellung.

v2.05

`cdfont=heavybold/ultrabold`

Die fetten Schriften werden stärker hervorgehoben. Es kommt **Open Sans Semi-Bold** respektive **Open Sans Bold** bei erhöhter Schriftstärke (`cdfont=heavy`) zum Einsatz.

Die folgende Tabelle verdeutlicht die Möglichkeiten zur Kombination der Einstellungen:

	<code>cdfont=normalbold</code>		<code>cdfont=heavybold</code>	
	<code>\mdseries</code>	<code>\bfseries</code>	<code>\mdseries</code>	<code>\bfseries</code>
<code>cdfont=true</code>	Open Sans Light	Open Sans Regular	Open Sans Light	Open Sans Semi-Bold
<code>cdfont=heavy</code>	Open Sans Regular	Open Sans Semi-Bold	Open Sans Regular	Open Sans Bold

Weiterhin lässt sich die Gestalt der Ziffern anpassen, wobei zwischen Versalziffern und Mediävalziffern gewählt werden.

v2.06

`cdfont=liningfigures/normalfigures`

Mit dieser Einstellung kommen dicktengleiche/äquidistant/proportionale Versalziffern oder auch Majuskelziffern (**1234567890**) zum Einsatz. Mit `\oldstylenums{<Ziffer(n)>}` lassen sich Mediävalziffern oder auch Minuskelziffern explizit setzen. Dieses Verhalten ist standardmäßig aktiviert.

v2.06

`cdfont=oldstylefigures/osf/oldnumbers`

Für den Fließtext werden nichtproportionale Mediävalziffern oder auch Minuskelziffern (**o123456789**) standardmäßig verwendet.

Außerdem dient die Option `cdfont=<Einstellung>` als erweiterte Schnittstelle für den Anwender, um zusätzliche Einstellungen für die Schriftnutzung vornehmen zu können, für welche normalerweise separate Optionen vorgesehen sind. Diese Möglichkeiten werden folgend kurz erläutert und auf die tatsächlich zugrunde liegende Option verweisen.

Die genutzten Mathematikschriften können mit folgenden Werte beeinflusst werden:

`cdfont=nomath/nocdmath`

Diese Einstellung deaktiviert die **Open Sans** für den mathematischen Satz. Es werden die  $\LaTeX$ -Standardschriften verwendet und es lassen sich beliebige Schriftpakete für den Mathematikmodus nutzen, *siehe* `cdmath=false`.

`cdfont=math/cdmath`

Die **Open Sans** wird für die Mathematikschriften genutzt, *siehe* `cdmath=true`.

`cdfont=upgreek/uprightgreek`

Griechische Groß- und Kleinbuchstaben sind aufrecht, *siehe* `slantedgreek=false`.

`cdfont=slgreek/slantedgreek`

Die Ausgabe der griechischen Glyphen erfolgt kursiv, *siehe* `slantedgreek=true`.

v2.06

`cdfont=texgreek/standardgreek`

Es kommt der  $\LaTeX$ -Standard zum Einsatz, bei dem griechische Majuskeln aufrecht und Minuskeln kursiv ausgegeben werden, *siehe* `slantedgreek=standard`.

v2.02

Für die TUD-Script-Klassen gibt es bei den tudheadings-Seitenstilen folgende Einstellmöglichkeiten für die Schriftart im Querbalken:

`cdfont=nohead/noheadfont`

Bei deaktivierter Hausschrift für den Fließtext können diese ebenfalls im Querbalken deaktiviert werden, *siehe* `cdhead=false`.

`cdfont=head/lighthouse/lightfonthead/noheavyfonthead`

Für den Querbalken der Kopfzeile wird – unabhängig von der Verwendung der Hausschrift im Fließtext – **Open Sans** in normaler Schriftstärke verwendet, *siehe* `cdhead=true`.

`cdfont=heavyhead/heavyfonthead`

Im Querbalken wird – unabhängig von der Dokumentschrift – **Open Sans** in erhöhter Stärke genutzt, *siehe* `cdhead=heavy`.

v2.06

`ttfont=<Einstellung>` Voreinstellung: `true`

Bei aktivierter Option `cdfont` lässt sich zusätzlich die verwendete Schreibmaschinenschrift einstellen, wofür selbige aus einem der Pakete **roboto-mono** oder **lmodern** genutzt wird und insbesondere die gewählten Einstellungen für die Schriftstärke der Option `cdfont` beachtet werden. Ein direktes Laden des jeweiligen Paketes durch den Benutzer ist nicht notwendig.

`ttfont=false/off/no`

Es findet keine Anpassung Schreibmaschinenschrift (`\ttfamily`) statt, mögliche Änderungen können durch den Anwender erfolgen.

`ttfont=true/on/yes/roboto/roboto-mono` (Säumniswert)

Sollte das Paket **roboto-mono** installiert sein, wird die darin definierte Schreibmaschinenschrift genutzt. Diese ist serifenlos und liegt in einer Vielzahl unterschiedlicher Schriftschnitte vor, weshalb sich diese ideal mit der **Open Sans** kombinieren lässt.

`ttfont=lmodern/lmtt/lm`

Es wird explizit die Schreibmaschinenschrift aus **lmodern** aktiviert.

### Auszeichnungen im Text

Unabhängig davon, welche Schriftfamilie im Dokument verwendet wird, können die Schriften des Corporate Designs jederzeit mit einem der hier aufgeführten Textschalter oder Textkommandos innerhalb des Dokumentes genutzt werden. Ein Textschalter wirkt sich – wenn er nicht lokal durch eine Gruppe oder Umgebung begrenzt wird – global auf das Dokument aus, wie etwa beispielsweise `\bfseries`. Bei einem Textkommando hingegen kommt die Schriftart nur für das nachfolgend angegebene Argument zum Einsatz, wie bei `\textbf{<Text>}`.

v2.04

`\cdfont{<Schriftart>}`  
`\textcd{<Schriftart>}{<Text>}`

Diese beiden Befehle dienen zur gezielten Aktivierung einer Schriftart des Corporate Designs in Stärke und Schnitt. Hierbei entspricht `\cdfont` einem Textschalter und ändert die verwendete Schriftart unverzüglich im aktuellen Geltungsbereich auf `<Schriftart>`, wohingegen `\textcd` als Textkommando fungiert und den im zweiten Argument gegebenen `<Text>` in `<Schriftart>` setzt ohne dabei die Dokumentschriftart selbst zu ändern.

Für die Schriftauswahl muss im ersten Argument die Bezeichnung der gewünschten Schriftart angegeben werden. Mögliche Werte sind der nachfolgenden Tabelle zu entnehmen. Der Vorsatz `Open Sans` für die `<Schriftart>` im Argument ist dabei optional. Ebenso sind weder Leerzeichen noch die passende Groß- und Kleinschreibung notwendig. Für die Wahl der Schriftstärke ist allein die Bezeichnung `Light/Regular/Semi-Bold/Bold/Extra-Bold` ausreichend.

Anstelle des Suffix' *Italic* ist auch die Nutzung von *Oblique* oder *Slanted* als Alias für die kursiven Schriftlagen möglich.

Schriftart/Bezeichnung	Schalter	Textkommando
Open Sans Light	<code>\cdfontln</code>	<code>\textcdln{&lt;Text&gt;}</code>
Open Sans Regular	<code>\cdfontrn</code>	<code>\textcdrn{&lt;Text&gt;}</code>
<b>Open Sans Semi-Bold</b>	<code>\cdfontsn</code>	<code>\textcdsn{&lt;Text&gt;}</code>
<b>Open Sans Bold</b>	<code>\cdfontbn</code>	<code>\textcdbn{&lt;Text&gt;}</code>
<b>Open Sans Extra-Bold</b>	<code>\cdfontxn</code>	<code>\textcdxn{&lt;Text&gt;}</code>
<i>Open Sans Light Italic</i>	<code>\cdfontli</code>	<code>\textcdli{&lt;Text&gt;}</code>
<i>Open Sans Regular Italic</i>	<code>\cdfontri</code>	<code>\textcdri{&lt;Text&gt;}</code>
<b><i>Open Sans Semi-Bold Italic</i></b>	<code>\cdfontsi</code>	<code>\textcdsi{&lt;Text&gt;}</code>
<b><i>Open Sans Bold Italic</i></b>	<code>\cdfontbi</code>	<code>\textcdbi{&lt;Text&gt;}</code>
<b><i>Open Sans Extra-Bold Italic</i></b>	<code>\cdfontxi</code>	<code>\textcdxi{&lt;Text&gt;}</code>

Alternativ zu `\cdfont` und `\textcd` werden für jeden Schriftschnitt auch explizite Textschalter und -kommandos bereitgestellt, bei denen das Argument für die Bezeichnung der Schriftart entfällt. Diese sind in der zweiten und dritten Spalte der Tabelle aufgeführt.

## 2.1.2 Schriften für den Mathematiksatz

v2.03

`cdmath=<Ein-Aus-Wert>`

Voreinstellung: `true` | `cdfont=false`: `false`

Diese Option dient zur Anpassung der Mathematikschriften. Wird diese aktiviert, so werden zur Hausschrift passende Glyphen im Mathematikmodus genutzt. Normale sowie fette Schriftstärke werden zu *Beginn des Dokuments* abhängig von der zu diesem Zeitpunkt aktiven Einstellung für die Schriften des Fließtexts (*siehe* `cdfont`) definiert. Auf fette Mathematikschriften kann im Dokument mit `\boldmath` umgeschaltet werden, eine Anpassung der Einstellungen ist durch `\TUOptions` möglich. Gültige Werte für die Option `cdmath` sind:

`cdmath=false/off/no`

Es werden die normalen  $\text{\LaTeX}$ -Serifenschriften beziehungsweise die Schriften beliebig nutzbarer Pakete für den Mathematiksatz verwendet.

`cdmath=true/on/yes` (Säumniswert)

Im Mathematikmodus kommt **Open Sans** sowohl für lateinische als auch griechische Buchstaben zum Einsatz.

v2.06

`slantedgreek=<Ein-Aus-Wert>`

Voreinstellung: `true`

Die Option ändert die standardmäßige Neigung der griechischen Großbuchstaben im Mathematikmodus bei der Verwendung der Standardbefehle für griechische Buchstaben.

`slantedgreek=false/off/no`

Die griechischen Glyphen werden im Mathematiksatz allesamt aufrecht gesetzt.

`slantedgreek=true/on/yes` (Säumniswert)

Alle griechischen Buchstaben werden im Mathematikmodus kursiv ausgegeben.

`slantedgreek=standard/latex`

Die Ausgabe entspricht mit aufrechten Majuskeln und kursiven Minuskeln der griechischen Lettern dem  $\text{\LaTeX}$ -Standard beim mathematischen Satz.

<code>\Gamma</code>
<code>\Delta</code>
<code>\Theta</code>
<code>\Lambda</code>
<code>\Xi</code>
<code>\Pi</code>
<code>\Sigma</code>
<code>\Upsilon</code>
<code>\Phi</code>
<code>\Psi</code>
<code>\Omega</code>
<code>\alpha</code>
<code>\beta</code>
<code>\gamma</code>
<code>\delta</code>
<code>\epsilon</code>
<code>\zeta</code>
<code>\eta</code>
<code>\theta</code>
<code>\iota</code>
<code>\kappa</code>
<code>\lambda</code>
<code>\mu</code>
<code>\nu</code>
<code>\xi</code>
<code>\pi</code>
<code>\varpi</code>
<code>\rho</code>
<code>\varrho</code>
<code>\sigma</code>
<code>\varsigma</code>
<code>\tau</code>
<code>\upsilon</code>
<code>\phi</code>
<code>\varphi</code>
<code>\chi</code>
<code>\psi</code>
<code>\omega</code>

Befehl (aufrecht)	Symbol	Befehl (kursiv)	Symbol
<code>\upGamma</code>	$\Gamma$	<code>\itGamma</code>	$\Gamma$
<code>\upDelta</code>	$\Delta$	<code>\itDelta</code>	$\Delta$
<code>\upTheta</code>	$\Theta$	<code>\itTheta</code>	$\Theta$
<code>\upLambda</code>	$\Lambda$	<code>\itLambda</code>	$\Lambda$
<code>\upXi</code>	$\Xi$	<code>\itXi</code>	$\Xi$
<code>\upPi</code>	$\Pi$	<code>\itPi</code>	$\Pi$
<code>\upSigma</code>	$\Sigma$	<code>\itSigma</code>	$\Sigma$
<code>\upUpsilon</code>	$\Upsilon$	<code>\itUpsilon</code>	$\Upsilon$
<code>\upPhi</code>	$\Phi$	<code>\itPhi</code>	$\Phi$
<code>\upPsi</code>	$\Psi$	<code>\itPsi</code>	$\Psi$
<code>\upOmega</code>	$\Omega$	<code>\itOmega</code>	$\Omega$
<code>\upalpha</code>	$\alpha$	<code>\italpha</code>	$\alpha$
<code>\upbeta</code>	$\beta$	<code>\itbeta</code>	$\beta$
<code>\upgamma</code>	$\gamma$	<code>\itgamma</code>	$\gamma$
<code>\updelta</code>	$\delta$	<code>\itdelta</code>	$\delta$
<code>\upepsilon</code>	$\epsilon$	<code>\itepsilon</code>	$\epsilon$
<code>\upzeta</code>	$\zeta$	<code>\itzeta</code>	$\zeta$
<code>\upeta</code>	$\eta$	<code>\iteta</code>	$\eta$
<code>\uptheta</code>	$\theta$	<code>\ittheta</code>	$\theta$
<code>\upiota</code>	$\iota$	<code>\itiota</code>	$\iota$
<code>\upkappa</code>	$\kappa$	<code>\itkappa</code>	$\kappa$
<code>\uplambda</code>	$\lambda$	<code>\itlambda</code>	$\lambda$
<code>\upmu</code>	$\mu$	<code>\itmu</code>	$\mu$
<code>\upnu</code>	$\nu$	<code>\itnu</code>	$\nu$
<code>\upxi</code>	$\xi$	<code>\itxi</code>	$\xi$
<code>\upi</code>	$\pi$	<code>\itpi</code>	$\pi$
<code>\upvarpi</code>	$\omega$	<code>\itvarpi</code>	$\omega$
<code>\uprho</code>	$\rho$	<code>\itrho</code>	$\rho$
<code>\upvarrho</code>	$\rho$	<code>\itvarrho</code>	$\rho$
<code>\upsigma</code>	$\sigma$	<code>\itsigma</code>	$\sigma$
<code>\upvarsigma</code>	$\varsigma$	<code>\itvarsigma</code>	$\varsigma$
<code>\uptau</code>	$\tau$	<code>\ittau</code>	$\tau$
<code>\upupsilon</code>	$\upsilon$	<code>\itupsilon</code>	$\upsilon$
<code>\upphi</code>	$\phi$	<code>\itphi</code>	$\phi$
<code>\upvarphi</code>	$\varphi$	<code>\itvarphi</code>	$\varphi$
<code>\upchi</code>	$\chi$	<code>\itchi</code>	$\chi$
<code>\uppsi</code>	$\psi$	<code>\itpsi</code>	$\psi$
<code>\upomega</code>	$\omega$	<code>\itomega</code>	$\omega$

Unabhängig von der Option `slantedgreek` können sowohl kursive als auch aufrechte griechischen Glyphen im Mathematikmodus mit diesen Befehlen direkt verwendet werden. Dies ist nützlich, um zwischen *kursiven Variablen* und *aufrechten Konstanten* zu unterscheiden. Weiterhin kann anstelle von `\up{<Glyph>}` oder `\it{<Glyph>}` auch der Präfix `\other{<Glyph>}` genutzt werden. Damit wird – abhängig von der Option `slantedgreek` – die komplementäre Schriftlage der angegebenen Glyphe gesetzt.

## Zusätzliche Hinweise zum Mathematiksatz

Unter Umständen werden zusätzliche Symbole für den Mathematiksatz benötigt. Sehr oft kommt hierfür das Paket **amssymb** zum Einsatz. Dieses stellt zahlreiche zusätzliche für die  $\text{\LaTeX}$ -Standardschriften zur Verfügung. Diese sind in Verbindung mit der **Open Sans** aus typografischer Sicht jedoch keine ideale Lösung. Die Symbole aus dem Paket **mdsymbol** passen wesentlich besser zu besagter Schriftfamilie, weshalb diesem Paket der Vorzug gegeben werden sollte, falls die Schriften des Corporate Designs für den mathematischen Satz genutzt werden. Weitere Hinweise zum typografisch guten Mathematiksatz sind außerdem in Kapitel 7 zu finden.

### 2.1.3 Vertikaler Leerraum in Abhängigkeit der Schriftgröße

Bei den TUD-Script-Klassen sind im Normalfall mehrere Längen von der für das Dokument gewählten Schriftgröße abhängig – im Gegensatz zu KOMA-Script. Dies hat den großen Vorteil, dass bei einer Änderung der Schriftgröße die folgend genannten Befehle respektive Längen nicht separat durch den Anwender angepasst werden müssen, um weiterhin sinnvoll verwendet werden zu können.

Die Anpassung an die Schriftgröße erfolgt sowohl die dehnbaren Längen `\bigskipamount`, `\medskipamount` und `\smallskipamount`, welche von den Befehlen `\bigskip`, `\medskip` beziehungsweise `\smallskip` für das Einfügen vertikaler Abstände genutzt werden, als auch die beiden Längen `\abovecaptionskip` und `\belowcaptionskip` für den Abstand zwischen einem Gleitobjekt – beispielsweise eine Abbildung oder eine Tabelle – und dessen mit `\caption` gesetzten Beschreibung. Außerdem wird die Länge `\columnsep` als Maß für den horizontalen Abstand der einzelnen Textspalten im zwei- oder mehrspaltigen Layout, wie es beispielsweise mit dem Paket **multicol** erzeugt werden kann, in Relation zur Schriftgröße sinnvoll festgelegt.

Die verwendete Schriftgröße kann durch den Anwender über die KOMA-Script-Option `fontsize=<Schriftgröße>` festgelegt werden. *Dabei ist zu beachten, dass diese immer als Klassenoption angegeben werden sollte.* Weitere Hinweise zur Wahl der passenden Schriftgröße sind außerdem in Abschnitt 3.1.1 zu finden.

Achtung!

v2.05

`relspacing=<Ein-Aus-Wert>` Voreinstellung: `true`

Mit der Option `relspacing=<Ein-Aus-Wert>` lässt sich die zuvor beschriebene Schriftgrößenabhängigkeit sowohl für vertikalen Leerraum zwischen zwei Absätzen oder bei Beschriftungen als auch für den horizontalen Abstand zwischen den Textspalten im mehrspaltigen Layout anpassen.

`relspacing=false/off/no/absolute/standard`

Die besagten Längen werden nicht angepasst, passende Werte sollten bei einer Änderung der Schriftgröße durch den Anwender gewählt werden.

`relspacing=true/on/yes/relative/fontsize` (Säumniswert)

In Abhängigkeit von der gewählten Schriftgröße werden die zuvor genannten Längen automatisch festgelegt.

## 2.2 Das Layout des Corporate Designs

### 2.2.1 Der Satzspiegel

v2.03  
v2.05

`cdgeometry=<Einstellung>` Voreinstellung: `true, restricted` | `cd=false: false`

Diese Option ist für die Aufteilung beziehungsweise die Berechnung des Satzspiegels verantwortlich. Das Maß der Seitenränder ist im Corporate Design fest vorgegeben und wird standardmäßig von den TUD-Script-Klassen eingehalten. Allerdings lassen sich die Seitenränder anpassen, um beispielsweise einen vernünftigen doppelseitigen Satz zu ermöglichen.<sup>15</sup> Des Weiteren besteht die Möglichkeit, auf das KOMA-Script-Standardverhalten zurückzufallen und die Satzspiegelberechnung durch das Paket **typearea** vornehmen zu lassen. Hier hat insbesondere die Klassenoption `DIV=<Einstellung>` maßgeblichen Einfluss auf den Satzspiegel.

v2.05

Mehr dazu ist in der Dokumentation von KOMA-Script<sup>16</sup> zu finden. Zusätzlich lässt sich mit `cdgeometry=custom` der Satzspiegel durch den Anwender (fast) beliebig festlegen.

`cdgeometry=false/off/no`

Die Satzspiegelberechnung erfolgt via **typearea**, die Vorgaben des Corporate Designs bezüglich der Seitenränder werden ignoriert.

`cdgeometry=true/on/yes/asymmetric/cd` (Säumniswert)

Die Seitenränder werden im asymmetrischen Stil des Corporate Designs fest definiert und auch für den doppelseitigen Satz (Klassenoption `twoside=true`) genutzt.<sup>17</sup>

`cdgeometry=symmetric/centred/centered`

Der Satzspiegel wird im einseitigen sowie doppelseitigen Satz auf der Seite zentriert.<sup>18</sup>

`cdgeometry=twoside/balanced`

Diese Einstellung aktiviert den doppelseitigen Satz (`twoside=true`) und ändert den Satzspiegel derart, dass die Ränder der inneren Seiten schmaler sind als die der äußeren.<sup>19</sup>

Achtung!

*Der so erzeugte Satzspiegel ist jedoch unvorteilhaft, da das Logo der Technischen Universität Dresden sehr nah am inneren Seitenrand des Dokumentes gesetzt wird und folglich auf rechten respektive ungeraden Seiten sehr weit an den Blattrand rückt.* Dieses Problem kann – bei **tudscrbook** sowie **tudscrreprt** – prinzipiell gelöst werden, indem Titel, Teile und Kapitel über das Aktivieren der KOMA-Script-Option `open=left` immer auf einer linken Seite beginnen, was allerdings aus typografischer Sicht eher unüblich ist.

v2.05

Mit den folgenden Werten lässt sich einstellen, in welcher Variante der Satzspiegel nach dem Corporate Design der Technischen Universität Dresden erstellt werden soll.

`cdgeometry=restricted`

Der Satzspiegel entspricht den expliziten Vorgaben des Corporate Designs.

`cdgeometry=adapted`

Laut dem Handbuch zum Corporate Design werden für Papierformate zwischen DIN A6 und DIN A4 „im Interesse größter Einheitlichkeit die Maßverhältnisse über einen größeren Formatbereich hinweg ‚eingefroren‘.“ Dies kann jedoch zu schlecht nutzbaren Satzspiegeln führen. Mit dieser Einstellung kann die Fixierung deaktiviert und der äquivalente Satzspiegel beispielsweise für das Format DIN A5 aktiviert werden. Bei Formaten außerhalb des fixierten Bereichs hat diese Einstellung keinen Einfluss.

<sup>15</sup>Hierbei sollte der innere Rand schmaler als der äußere sein

<sup>16</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

<sup>17</sup>links: 30 mm, rechts: 20 mm, oben: 20 mm, unten: 30 mm

<sup>18</sup>links: 25 mm, rechts: 25 mm, oben: 20 mm, unten: 30 mm

<sup>19</sup>innen: 20 mm, außen: 30 mm, oben: 20 mm, unten: 30 mm

`cdgeometry=calculated`

Der Satzspiegel wird anhand der Referenzmaße für das Format DIN A4 für das eingestellte Papierformat *skaliert*. Die eigentlich definierten diskreten Maße bei unterschiedlichen Gestaltungshöhen werden ignoriert.

v2.05

Da es häufig sehr restriktive – wenn auch meistens völlig willkürliche – Vorgaben für die Seitenränder gibt, besteht außerdem die Möglichkeit, diese weitestgehend manuell einzustellen.

`cdgeometry=custom`

Für die Festlegung der Seitenränder ist das Paket **geometry** maßgeblich verantwortlich. Der Anwender kann mit den beiden, durch das Paket bereitgestellten Befehlen `\geometry` und `\newgeometry` den Satzspiegel selbst definieren. Für Hinweise zur Verwendung der Befehle sollte die Dokumentation von **geometry** genutzt werden.

v2.03

`extrabottommargin=<Höhe>` Voreinstellung: `0pt`

Mit dieser Option kann die Größe des unteren Seitenrandes angepasst werden, wenn der Satzspiegel des Corporate Designs verwendet wird (*siehe* `cdgeometry`). Insbesondere für den Fall, dass bei Seiten im Stil `tudheadings` im Fußbereich entweder mit `\footlogo` Drittlogos verwendet werden und diese über das optionale Argument oder via `footlogoheight` über die Standardhöhe hinaus vergrößert wurden oder mit `\footcontent` ein übergroßer Inhalt angegeben wurde, kann dieser unter Umständen etwas zu klein sein. Mit der Option `extrabottommargin` wird der Fußbereich durch positive Werte vergrößert, negative Werte verkleinern diesen entsprechend.

Alternativ zur Option `extrabottommargin` kann auch die Einstellung `cdfoot=<Höhe>` mit einem Längenwert verwendet werden. Dabei spielt es für beide Optionen keine Rolle, ob eine  $\text{\LaTeX}$ -Länge, ein  $\text{\TeX}$ -Abstand oder eine  $\text{\TeX}$ -Ausdehnung als Länge bei der Wertzuweisung verwendet wird.

### Bindekorrektur

Im Zusammenhang mit den Seitenrändern oder besser dem Satzspiegel ist die durch das Paket **typearea** zur Verfügung gestellte KOMA-Script-Option `BCOR=<Längenwert>` zu erwähnen. Mit dieser kann bei der Satzspiegelberechnung ein Heftrand respektive eine Bindekorrektur berücksichtigt werden. Durch die TUD-Script-Klassen wird der mit dieser Option angegebene Wert auch an das Paket **geometry** weitergereicht, sodass der Benutzer unabhängig von der Satzspiegelgestaltung (`cdgeometry`) die Option `BCOR` nutzen kann. So lässt sich eine Bindekorrektur von beispielsweise 5 mm mit der *Klassenoption* `BCOR=5mm` festlegen.

Eine Anpassung der Bindekorrektur hat natürlich *immer* eine Änderung der verfügbaren Breite des Textbereichs zur Folge hat und führt somit zwingend zu einer Anpassung des Satzspiegels. Da die Bindekorrektur jedoch abhängig von der Höhe des Buchblocks gewählt werden sollte, welche letztendlich erst mit dem Druck des fertiggestellten Dokumentes bestimmt werden kann, muss diese zu Beginn abgeschätzt werden.

**Beispiel:** Als Faustregel gilt, dass die erforderliche Bindekorrektur in etwa der halben Höhe des Buchblocks entsprechen sollte. Dessen Höhe wiederum ist abhängig von der Anzahl der Seiten sowie der Dichte des verwendeten Papiers. Wird normales Papier mit einer Dichte von  $80\text{ g/m}^2$  verwendet, so entsprechen 100 Blatt in etwa einer Höhe von 10 mm, bei  $100\text{ g/m}^2$  ca. 12 mm. Dementsprechend wäre eine Bindekorrektur von `BCOR=5mm` beziehungsweise `BCOR=6mm` bei diesem Beispiel zu wählen.

## Kopf- und Fußzeile im Zusammenspiel mit dem Satzspiegel

Da im Corporate Design nicht festgelegt ist, wie die Gestaltung der Kopf- und Fußzeilen in einer wissenschaftlichen Arbeit auszuführen ist, bleibt dem Nutzer dafür eine gewisse Freiheit. Dafür sollte idealerweise das zu KOMA-Script gehörige Paket `sclayer-scrpage` genutzt werden.

In der Dokumentation zu `typearea` wird auch darauf eingegangen, wann Kopf- und Fußzeile bei der Satzspiegelkonstruktion entweder dem Rand oder dem Textkörper zugeschlagen werden sollten. Dies sollte bei der Erstellung eigener Kopf- und Fußzeilen beachtet werden. Die Einstellung dafür erfolgt mit den beiden KOMA-Script-Optionen `headinclude=<Ein-Aus-Wert>` sowie `footinclude=<Ein-Aus-Wert>`. Diese können – unabhängig von der Einstellung zur Satzspiegelgestaltung durch die Option `cdgeometry` – verwendet werden.

### 2.2.2 Die Gestalt von Titel, Umschlagseite, Teilen sowie Kapiteln & Co.

Falls die nachfolgend beschriebene Option `cd=<Einstellung>` aktiviert ist, werden einige, spezielle Seiten im prägnanten Stil mit dem Logo der Technischen Universität Dresden und der dazugehörigen Kopfzeile mit Querbalken gesetzt. Dies betrifft insbesondere die Umschlagseite und den Titel aus Abschnitt 2.2.4, die Teileseiten in Abschnitt 2.2.5 sowie die Kapitelseiten in Abschnitt 2.2.6. Mit den `tudheadings`-Seitenstilen oder der `tudpage`-Umgebung können weitere Seiten in diesem Stil erzeugt werden. Wird das Paket `tudscrsupervisor` verwendet und mit den bereitgestellten Befehlen oder Umgebungen eine Aufgabenstellung, ein Gutachten oder ein Aushang erstellt, so erscheinen auch diese in besagtem Seitenstil.

v2.03  
v2.04

`cd=<Einstellung>` Voreinstellung: `true`

Mit dieser Option wird festgelegt, ob und wie das Corporate Design der Technischen Universität Dresden im gesamten Dokument verwendet wird. Sie hat Einfluss auf die Ausprägung von Titel-, Teil-, und Kapitelseiten sowie die Überschriften der weiteren Gliederungsebenen.

`cd=false/off/no`

Diese Einstellung erzeugt das Standard-Verhalten der KOMA-Script-Klassen, es wird kein Corporate Design genutzt.

`cd=true/on/yes/nocolor/monochrome` (Säumniswert)

Das Layout für Titel-, Teil- und Kapitelseiten ist im Corporate Design, es wird schwarze Schrift für Titel, Teil- und Kapitelüberschriften verwendet. Die Ausprägung des Seitenkopfes ist abhängig von der Option `cdhead`.

`cd=lightcolor/pale`

Die Einstellung entspricht weitestgehend der Option `cd=true`, allerdings wird die primäre Hausfarbe HKS41 für den Kopf des `tudheadings`-Seitenstils und Überschriften genutzt.

v2.04

`cd=barcolor`

Zusätzlich zur vorherigen Einstellung wird außerdem der Querbalken farbig abgesetzt.

v2.03

`cd=bicolor/bichrome`

Der Kopf wird mit einem farbigen Hintergrund in der Hausfarbe gesetzt, auch der Querbalken wird farbig hinterlegt. Für die Überschriften wird die primären Hausfarbe verwendet.

`cd=color`

Der Titel sowie Teil- und Kapitelseiten werden allesamt farbig gestaltet, der Seitenkopf wird in der primären Hausfarbe HKS41 gesetzt, der Querbalken erhält Linien als Begrenzung.

v2.03

`cd=fullcolor/full`

Entspricht der vorherigen Einstellung, allerdings wird der Querbalken nicht durch Linien begrenzt sondern farbig hinterlegt.



v2.06

`headings=<Einstellung>`

Voreinstellung: heavy

Die Option `headings=<Einstellung>` wird bereits von KOMA-Script definiert und dient unter anderem zum Festlegen der Größe von Überschriften. Diese wird von TUD-Script erweitert, um im Layout des Corporate Designs die Schriftstärke der Überschriften anpassen zu können.

`headings=light/normalbold`

Die Überschriften werden in fester Schrift gesetzt.

`headings=heavy/ultrabold`

Für Überschriften wird die extra-fette Schriftstärke verwendet.

Die Einstellungen gelten für die Gliederungsebenen bis einschließlich `\subsubsection`.

## Individuelle Einstellungen für einzelne Elemente des Layouts

Das Verhalten aller für das Layout relevanten Elemente wird von der eben zuvor erläuterten Option `cd=<Einstellung>` bestimmt. Dies betrifft zum einen sowohl den Titel (`\maketitle`) als auch die Umschlagseite (`\makecover`) und zum anderen alle Teileseiten (`\part`, `\addpart`) und Kapitelseiten (`\chapter`, `\addchap`) sowie alle darunter liegenden Gliederungsebenen.

Soll ein bestimmtes Element des Layouts abweichend von der allgemeinen Einstellung für das gesamte Dokument erscheinen, so kann eine der folgenden Optionen genutzt werden, um dieses individuell anzupassen und die mit `cd=<Einstellung>` eingestellten Vorgaben für das jeweilige Element zu überschreiben.<sup>20</sup> Die gültigen Wertzuweisungen für die einzelnen Elemente entsprechend dabei den möglichen Werten für die Option `cd=<Einstellung>`.

Zu beachten ist dabei, dass die verwendete Schrift für die Elemente des Layouts abhängig von der Option `cdfont=<Einstellung>` ist. Für Titel- und Umschlagseite kann diese direkt über das optionale Argument von `\maketitle` respektive `\makecover` geändert werden.

v2.03

v2.04

`cdtitle=<Einstellung>`

Mit der Option `cdtitle` kann die allgemeine Einstellung für den Titel überschrieben werden. Es kann zwischen dem normalen (`cdtitle=false`) und dem im Corporate Design umgeschaltet werden. Die neue Titelseite unterstützt alle durch KOMA-Script definierten Befehle für den Titel.<sup>21</sup> Zusätzlich werden viele neue Felder definiert, welche vor allem für eine wissenschaftliche Arbeit von Relevanz sind. Genaueres dazu ist in Abschnitt 2.2.4 nachzulesen. Unabhängig von der gewählten Variante der Titelseite wird diese immer mit `\maketitle` erzeugt.

v2.02

v2.03

v2.04

`cdcover=<Einstellung>`

Die TUD-Script-Klassen führen zusätzlich den Befehl `\makecover` ein, mit dem sich neben dem Titel eine separate Umschlagseite erzeugen lässt. Diese ist in ihrer Gestalt der Titelseite sehr ähnlich, wird normalerweise jedoch in einem anderen Satzspiegel als dem des Buchblocks gesetzt. Mit der Option `cdcover` kann – unabhängig von `cd=<Einstellung>` – das Aussehen der Umschlagseite geändert werden. Wird `cdcover=false` gewählt, entspricht die Umschlagseite dem originalen KOMA-Script-Titel. Die Verwendung des Befehls `\makecover` sowie die dazugehörigen Parameter werden detailliert in Abschnitt 2.2.4 erläutert.

<sup>20</sup>`cdtitle` für den Titel, `cdcover` für die Umschlagseite, `cdpart` für Teile, `cdchapter` für Kapitel sowie `cdsection` für alle darunter liegenden Gliederungsebenen

<sup>21</sup>`\extratitle{<Schmutztitel>}`, `\frontispiece{<Frontispiz>}` für eine Seite vor dem eigentlichen Haupttitel sowie `\titlehead{<Kopf>}`, `\subject{<Typisierung>}`, `\title{<Titel>}`, `\subtitle{<Untertitel>}`, `\author{<Autor>}`, `\and`, `\thanks{<Fußnote>}`, `\publishers{<Verlag>}` und `\date{<Datum>}` für den Titel selbst sowie `\uppertitleback{<Titelrückseitenkopf>}` und `\lowertitleback{<Titelrückseitenfuß>}` für die Rückseite des Titels sowie abschließend `\dedication{<Widmung>}` für eine Danksagung.

v2.03  
v2.04

`cdpart=<Einstellung>`

Für die Teileseiten kann der Wert des Schlüssels `cd=<Einstellung>` separat überschrieben und somit deren Layout respektive Erscheinungsbild beeinflusst werden, welches bei der Benutzung der Befehle `\part` und `\addpart` sowie deren Sternversionen verwendet wird. In Abschnitt 2.2.5 sind weitere Hinweise zur Teileseite im Corporate Design zu finden.

v2.03  
v2.04

`cdchapter=<Einstellung>`

Für Kapiteleiten kann der Schlüsselwert `cd=<Einstellung>` ebenfalls angepasst und damit das Layout respektive Erscheinungsbild geändert werden, das bei der Verwendung von `\chapter` beziehungsweise `\addchap` und den dazugehörigen Sternversionen genutzt wird. Weitere Hinweise zur Kapiteleite im Corporate Design sind in Abschnitt 2.2.6 zu finden.

v2.05

`cdsection=<Einstellung>`

Für Überschriften der weiteren Gliederungsebenen `\section`, `\subsection`, `\subsubsection` sowie `\paragraph` und `\subparagraph` werden in der primären Hausfarbe HKS41 ausgegeben, falls über die Option `cd=<Einstellung>` eine farbige Ausprägung des Layouts eingestellt wurde. Mit der Angabe von `cdsection=true` erscheinen die Überschriften der genannten Gliederungsebenen ohne Farbeinsatz.

**Beispiel:** Soll die Titelseite in Farbe, der Rest des Dokumentes allerdings in schwarzer Schrift gesetzt werden, so kann dies folgendermaßen erreicht werden:

```
\documentclass[cd=true,cdtitle=color]{<Dokumentklasse>}
```

## Die vertikale Position der Überschriften

v2.05

`pageheadingsvskip=<Längenwert>`  
`headingsvskip=<Längenwert>`

Mit diesen beiden Optionen kann die vertikale Position spezieller Überschriften verändert werden. Mit der Option `pageheadingsvskip` lässt sich sowohl der Titel auf einer Titelseite (KOMA-Script-Option `titlepage=true`) als auch die Überschriften von separaten Kapiteleiten (`chapterpage=true`) und Teilen vertikal verschieben. Demgegenüber kann mit der zweiten Option `headingsvskip`, sowohl den Titel des Titelpfades (KOMA-Script-Option `titlepage=false`) als auch die Kapitelüberschrift bei deaktivierter Kapiteleite (`chapterpage=false`) in ihrer vertikalen Position anzupassen.

Die zuvor genannten Überschriften werden normalerweise im Layout relativ tief im Textbereich gesetzt. Mit negativen Werten werden die Überschriften nach oben verschoben, wobei darauf geachtet werden sollte, dass diese sich danach noch innerhalb des Satzspiegels befinden. Positive Werte setzen die Überschriften dementsprechend tiefer.

## 2.2.3 Seiten im Stil des Corporate Designs

```
\faculty{<Fakultät>}  
\department{<Einrichtung>}  
\institute{<Institut>}  
\chair{<Lehrstuhl>}  
\extraheadline{<Textzeile>}
```

Für den Seitenstil des Corporate Designs der Technischen Universität Dresden charakteristisch ist die Kopfzeile mit dem prägnanten Querbalken. In dieser wird – falls angegeben – in fetter

Schrift die Fakultät ausgegeben, danach folgen durch Kommas getrennt die Einrichtung, das Institut und der Lehrstuhl beziehungsweise die Professur. Sollte der Platz in der ersten Zeile nicht ausreichen, erfolgt ein automatischer Zeilenumbruch.

In besonderen Ausnahmefällen erlaubt das Corporate Design die Angabe einer zusätzlichen zweiten beziehungsweise dritten Zeile unterhalb der Angaben des Bereichs an der Technischen Universität Dresden, welche weitere, frei wählbare Angaben enthält. Diese kann mit dem Befehl `\extraheadline{<Textzeile>}` definiert werden.

v2.02

```
tudheadings (Seitenstil)
plain.tudheadings (Seitenstil)
empty.tudheadings (Seitenstil)
```

Ein zentrales Element des Corporate Designs der Technischen Universität Dresden ist der prägnante Seitenkopf mit der Angabe von Fakultät (`\faculty`), Einrichtung (`\department`), Institut (`\institute`) und Lehrstuhl (`\chair`) im dazugehörigen Querbalken. Durch die Nutzung des Paketes **scrlayer-scrpage** lassen sich entweder einzelne Seiten oder auch ganze Dokumente sehr einfach in diesem Stil setzen. Dazu muss lediglich mit `\pagestyle{<Seitenstil>}` einer der Seitenstile geladen werden.

Allen Seitenstilen gemein ist der typische Kopf mit dem charakteristischen Querbalken, dessen Gestalt für *alle* Seitenstile gleichermaßen über die Option `cdhead` angepasst werden kann. Mit dem Befehl `\headlogo` lässt sich ein zusätzliches Zweitlogo im Kopfbereich ausgegeben. *Für die speziellen Layout-Elemente Titel und Umschlagseite sowie Teile- und Kapitelseiten wird die Einstellung von `cdhead` durch die Nutzung der Option `cd=<Einstellung>` überschrieben.*

Achtung!

Die Ausprägung des Fußes unterscheidet sich bei den einzelnen Seitenstilen. Dieser ist beim Seitenstil `empty.tudheadings` immer leer. Die beiden Stile – oder vielmehr das Seitenstil-Paar – `tudheadings` und `plain.tudheadings` übernehmen die Einstellungen für die Fußzeile aus der Anwenderschnittstelle von **scrlayer-scrpage**.<sup>22</sup> Wie die einzelnen Befehle zur Individualisierung der Fußzeile zu verwenden sind, kann der KOMA-Script-Anleitung<sup>23</sup> entnommen werden. Alternativ zu einer eigenen Definition der Fußzeile lässt sich außerdem die Option `cdfoot` verwenden. Zusätzlich kann über `\footcontent` ein freier Inhalt für den Fußbereich definiert werden, mit `\footlogo` ist die Ausgabe von einem oder mehreren Logos in diesem möglich. Die verwendete Schrift im Fußbereich wird durch das Schriftelement `tudheadings` festgelegt.

Sobald einer der definierten Stile mit `\pagestyle{<Seitentil>}` aktiviert wurde, sind die beiden Seitenstile `tudheadings` sowie `plain.tudheadings` zusätzlich unter den Namen `headings` respektive `plain` verwendbar. Dies hat den Vorteil, dass bei Optionen oder Befehlen, welche automatisch zwischen den beiden Seitenstilen `headings` und `plain` umschalten, durch die einmalige Auswahl von einem der `tudheadings`-Stilen nun zwischen diesen umgeschaltet wird.

Der Seitenstil `empty` erzeugt allerdings weiterhin eine komplett leere Seite. Soll eine Seite mit der prägnanten Kopfzeile der Technischen Universität Dresden und leerem Seitenfuß erschienen, so muss `\pagestyle{empty.tudheadings}` manuell aufgerufen werden. Um auf das normale Verhalten von KOMA-Script zurückzuschalten, muss mit `\pagestyle{<Seitentil>}` einer der beiden Seitenstile `scrheadings` oder `plain.scrheadings` aktiviert werden.

v2.03  
v2.04  
v2.05

```
cdhead=<Einstellung>   Voreinstellung: true,nocolor,textwidth
```

Mit dieser Option lassen sich für die `tudheadings`-Seitenstile sowohl die Gestalt des Logos sowie des Querbalkens als auch die darin verwendete Schrift beeinflussen. Die folgenden Werte

<sup>22</sup>Es können die Befehle `\lfoot`, `\cfoot` und `\rfoot` sowie `\lfoot`, `\cofoot` und `\rofoot` respektive `\lofoot`, `\cfoot` und `\ifoot` genutzt werden.

<sup>23</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

können für eine Anpassung der Schriftart im Balken verwendet werden:

`cdhead=false/off/no`

Sollte mit `cdfont=false` die Verwendung der Hausschrift im Stil des Corporate Designs der Technischen Universität Dresden deaktiviert worden sein, wird die Kopfzeile im Querbalken in den Serifenlosen der aktiven Schrift gesetzt. Ist die Hausschrift aktiviert, hat diese Einstellung keinen Einfluss.

`cdhead=true/on/yes/light/lightfont/noheavyfont` (Säumniswert)

Im Querbalken wird für `\faculty` **Open Sans Semi-Bold** verwendet, für `\department`, `\institute`, `\chair` und `\extraheadline` kommt Open Sans Light zum Einsatz.

`cdhead=heavy/heavyfont`

Der Inhalt von `\faculty` wird in **Open Sans Bold** gesetzt, für die restlichen Felder wird **Open Sans Regular** genutzt.

Bei der Ausprägung des Kopfes und des Querbalkens gibt es mehrere Varianten. Einerseits kann der Querbalken mit zwei Außenlinien dargestellt werden:

`cdhead=nocolor/monochrome`

Der Kopf und die Linien des Querbalkens erscheinen in schwarzer Farbe.

`cdhead=lightcolor/pale`

Sowohl Kopf als auch Querbalken werden in der primären Hausfarbe gesetzt.

Andererseits ist auch eine Darstellung mit mehr Farbeinsatz möglich, bei welcher der Querbalken und gegebenenfalls der ganze Seitenkopf farbig abgesetzt wird. Dabei erstreckt sich der Kopfbereich immer über die komplette Seitenbreite, *siehe* `cdhead=paperwidth`.

v2.04

`cdhead=barcolor`

Im Gegensatz zur vorherigen Einstellung wird der Querbalken mit farbigem Hintergrund verwendet, der darüber liegende Teil des Kopfes wird ohne farbigen Hintergrund gesetzt.

v2.03

`cdhead=bicolor/bichrome`

Die Kopfzeile wird farbig abgesetzt, wobei der Hintergrund des Logos und der Querbalken unterschiedlich ausfallen. Die Außenlinien der Querbalkens entfallen.

Für den Fall, dass der Querbalken lediglich mit zwei Außenlinien dargestellt wird, kann zusätzlich dessen Laufweite festgelegt werden:

`cdhead=textwidth/slim`

Der Querbalken im Kopf erstreckt sich nur über den Textbereich. Diese Einstellung ist insbesondere sinnvoll, wenn ein randloser Ausdruck technisch nicht möglich ist.

`cdhead=paperwidth/wide`

Die horizontale Ausdehnung des Querbalkens erstreckt sich über die komplette Seitenbreite bis an den Blattrand. Dieses Verhalten kann im farbigen Layout des Kopfes nicht deaktiviert werden.

v2.05

Neben den zuvor beschriebenen Möglichkeiten zur Gestaltung des Kopfbereiches kann auf allen Seiten mit aktiviertem `tudheadings`-Seitenstil unterhalb des Querbalkens das mit `\date` angegebene Datum rechtsbündig eingeblendet werden.

`cdhead=date/showdate`

Das eingestellte Datum wird rechts oberhalb vom Textbereich eingeblendet.

`cdhead=nodate/hidedate`

Es erscheint kein Datum zwischen Kopf- und Textbereich.

v2.03

`cdfoot=<Einstellung>` Voreinstellung: false

Die TUD-Script-Klassen sind – insbesondere aufgrund der Möglichkeit zur Verwendung des Paketes **scrlayer-scrpage** – bei der Gestaltung der Kopf- und Fußzeilen sehr flexibel und individuell anpassbar. Die Ausprägung und der Inhalt dieser ist nicht explizit durch das Corporate Design vorgegeben und können durch den Anwender beliebig gewählt und geändert werden.

Eine Möglichkeit für deren Gestaltung zeigt das Handbuch für das Corporate Design der Technischen Universität Dresden. Dieses wird ohne Kopf- und mit einer einfachen Fußzeile gesetzt, welche den aktuellen Kolumnentitel sowie die Paginierung enthält. Mit `cdfoot` kann diese Ausprägung aktiviert werden, was auch für dieses Anwenderhandbuch geschehen ist.

`cdfoot=false/off/no`

Die Kopf- und Fußzeilen zeigen Standardverhalten, zur manuellen Änderung sollte unbedingt das KOMA-Script-Paket **scrlayer-scrpage** verwendet werden.

`cdfoot=true/on/yes` (Säumniswert)

Die Fußzeilen des Dokumentes werden äquivalent zum Handbuch des Corporate Designs der Technischen Universität Dresden mit lebenden Kolumnentitel und Seitenzahl gesetzt, wobei im doppelseitigen Satz (Klassensoption `twoside=true`) die Paginierung außen platziert wird.

Wird beim Laden der Klasse respektive des Paketes **scrlayer-scrpage** die Option `manualmark` nicht explizit angegeben, so werden mit `cdfoot=true` über die KOMA-Script-Option `automark` auch gleichzeitig die automatischen Kolumnentitel aktiviert, welche als Marken die Titel der Gliederungsebenen verwendet. Genaueres dazu und der Möglichkeit, die Kolumnentitel manuell festzulegen, ist dem KOMA-Script-Handbuch<sup>24</sup> zu entnehmen.

v2.03

Sollte einer der `tudheadings`-Seitenstil aktiviert sein und es wird auf der erzeugten Seite ein farbiges Layout – beispielsweise der zweifarbige Kopf (`cdhead=bicolor`) oder ein farbiger Seitenhintergrund – genutzt, so kann auch die Fußzeile einen farbigen Hintergrund erhalten.

`cdfoot=nocolor/monochrome`

Der Fuß wird immer ohne farbigen Hintergrund gesetzt.

`cdfoot=color/bicolor/bichrome`

Die Fußzeile wird farbig abgesetzt, falls entweder der Kopf in einer farbigen Variante genutzt wird (*siehe* `cdhead`) oder eine Seite mit einem farbigen Hintergrund in der Hausfarbe (Titel oder Kapitelseite) aktiv ist.

Mit den Befehlen `\footlogo` auf Seite 30 und `\footcontent` auf Seite 30 können für den Fußbereich zusätzliche Inhalte definiert werden. Sollte der zur Verfügung stehende Platz hierfür nicht ausreichen, lässt sich dieser vergrößern.

`cdfoot=<Höhe>`

Wird der Option ein Längenwert übergeben, entspricht dies exakt der Verwendung von Option `extrabottommargin=<Höhe>`.

v2.04

`tudheadings` (Schriftelement)

Im Fußbereich der Seiten im `tudheadings`-Seitenstil wird das Schriftelement `tudheadings` verwendet. Dieses wirkt sich auf die Seitenzahlen, den Kolumnentitel und die mit `\footcontent` angegebenen Inhalte aus. Hierüber wird die Wahl der richtigen Schriftfarbe in Abhängigkeit vom Seitenhintergrund und den Einstellungen für die Optionen `cdhead` sowie `cdfoot` realisiert. Wie `tudheadings` angepasst werden kann, ist in Abschnitt 2.2.8 zu finden.

<sup>24</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

```
\headlogo[<Parameterliste>]{<Dateiname>}
```

Neben dem Logo der Technischen Universität Dresden darf zusätzlich ein Zweitlogo im Kopf verwendet werden. Dieses lässt sich mit diesem Befehl einbinden. Normalerweise wird es auf die Höhe der Erstlogos skaliert. Über das optionale Argument können weitere Formatierungsbefehle an den im Hintergrund verwendeten Befehl `\includegraphics` durchgereicht werden, um beispielsweise die Größe des Zweitlogos anzupassen. Welche Parameter angepasst werden können, ist der Dokumentation des **graphicx**-Paketes zu entnehmen.

Sollte die Option `ddc` aktiviert sein, wird das DRESDEN-concept-Logo nicht im Kopf sondern automatisch im Fuß gesetzt. Die Option `ddchead` setzt dieses auf jeden Fall im Kopf und überschreibt damit das mit `\headlogo` angegebene Zweitlogo.

v2.03

```
\footlogo[<Parameterliste>]{<Dateinamenliste>}  
\footlogosep
```

Laut den Richtlinien des Corporate Designs dürfen im Fußbereich weitere Logos erscheinen, beispielsweise von kooperierenden Unternehmen oder Sponsoren. Die Dateinamen der gewünschten Logos können als kommaseparierte Liste im obligatorischen Argument des Befehls `\footlogo` angegeben werden. Sollte tatsächlich nicht nur ein Dateiname sondern eine Liste übergeben worden sein, so wird bei der Ausgabe der Logos zwischen diesen jeweils der in `\footlogosep` gespeicherte Separator – standardmäßig `\hfill` – gesetzt. Dieser kann mit `\renewcommand*{\footlogosep}{. . .}` beliebig durch den Anwender angepasst werden. Der Separator wird auch gesetzt, falls in `<Dateinamenliste>` lediglich ein Komma verwendet wurde. Mit `\footlogo{, <Dateiname>,}` kann so beispielsweise ein Logo zentriert im Fuß gesetzt werden. *Dabei ist zu beachten, dass ein mit der Option `ddc` beziehungsweise `ddcfoot` gesetztes DRESDEN-concept-Logo im Fußbereich – im Gegensatz zur Verwendung von `\footcontent` – überlagert werden könnte. Hier muss der Anwender im Zweifel durch das Einfügen von Separatoren – sprich Kommas – im Argument von `\footlogo` etwas Formatierungsarbeit leisten.*

Achtung!

Das optionale Argument von `\footlogo` wird an `\includegraphics` weitergereicht. Dies geschieht für alle angegebenen Dateien aus der Liste gleichermaßen. Sollen für einzelne Logos individuelle Einstellungen vorgenommen werden, so sind die entsprechenden Parameter im obligatorischen Argument nach dem jeweiligen Dateinamen mit einem Doppelpunkt „:“ als Separator (`\footlogo{<Dateiname>:<Parameter>}`) zu übergeben, wobei diese *nach* den allgemeinen Einstellungen für alle Logos angewendet werden. Die möglichen Parameter und Werte für die optionalen Argumente sind der Dokumentation des **graphicx**-Paketes zu entnehmen.

v2.05

```
footlogoheight=<Längenwert>
```

Ohne die Angabe eines optionalen Argumentes bei `\footlogo` für die Größe werden alle Logos im Fuß auf die Höhe des Logos der Technischen Universität Dresden skaliert. Dies kann global für alle Logos geändert werden, indem die Option `footlogoheight=<Längenwert>` gesetzt wird. Sollte die Höhe des Fußbereiches nicht ausreichen, um alle Logos in der gewünschten Größe darstellen zu können, kann diese über `extrabottommargin=<Höhe>` beziehungsweise `cdfoot=<Höhe>` angepasst werden.

v2.04

```
\footcontent[<Anweisungen>]{<Inhalt>}[<Inhalt>]
```

v2.05

```
\footcontent* [<Anweisungen>]{<Inhalt>}[<Inhalt>]
```

Mit diesem Befehl kann beliebiger Inhalt entweder einspaltig oder zweiseitig im Fußbereich der `tudheadings`-Seitenstile gesetzt werden. In der Form `\footcontent{<Inhalt>}` wird der Inhalt über die komplette Textbreite im Fuß ausgegeben. Wird der Befehl jedoch in der zweiten Variante `\footcontent{<linker Inhalt>}[<rechter Inhalt>]` mit einem optionalen *nach*

dem obligatorischen Argument verwendet, so erscheint der Fußbereich zweispaltig, wobei der Inhalt aus dem ersten, obligatorischen Argument in der linken und der Inhalt aus dem zweiten, optionalen Argument entsprechend in der rechten Fußspalte gesetzt wird. Dabei wird ein etwaiges DRESDEN-concept-Logo, welches über die Option `ddc` respektive `ddcfoot` gesetzt wurde, beachtet und der für den Text zur Verfügung stehende Platz im Fuß reduziert. Mögliche Einstellungen des Paketes `scrlayer-scrpage` für den Fuß werden nicht berücksichtigt, hier kann es zu Überlagerungen der Inhalte kommen. Gleiches gilt für die Verwendung der Werte `true` und `false` für die Option `cdfoot`.

v2.05

Wird an das Argument für die linke oder die rechte Spalte lediglich ein Stern `*` übergeben, so bleibt der bis dahin definierte Inhalt in dieser Spalte erhalten. Beispielsweise kann die linke Fußbereichsspalte mit `\footcontent{<Inhalt>[*]}` angepasst werden ohne dabei den Inhalt der rechten Spalte zu verändern oder es ließe sich lediglich die verwendete Schrift des Fußbereichs bei gleichbleibendem Inhalt mit `\footcontent[<Anweisungen>]{*}{*}` anpassen.

Im Fußbereich wird für die Schrift das Schriftelement `tudheadings` verwendet. Dabei wird auch die Schriftgröße angepasst, wobei diese sich an der Kopfzeile orientiert. Zusätzlich können mit dem ersten optionalen Argument von `\footcontent` – vor der eigentlichen Ausgabe des Inhaltes – zusätzliche Schrifteinstellungen respektive `<Anweisungen>` ausgeführt werden. Soll die Definition des Inhalts im Fußbereich *ohne* eine automatische Anpassung der Schriftgröße erfolgen, so ist die Sternversion `\footcontent*` zu verwenden. Auch hier lässt sich gegebenenfalls das optionale Argument für die Schriftformatierung nutzen. Zu guter Letzt können bei der Änderung des Inhaltes im Fußbereich auch vorherige Anpassungen der Schrift unverändert bleiben, indem mit `\footcontent[*]{<Inhalt>[<Inhalt>]}` im optionalen Argument ein Stern verwendet wird.

v2.06

v2.02

```
ddc=<Einstellung>
ddchead=<Einstellung>
ddcfoot=<Einstellung>
```

Voreinstellung: `false`

Diese Option fügt das Logo von DRESDEN-concept entweder im Kopf oder Fuß der Seiten mit dem Stil `tudheadings` ein. Diese wird automatisch entweder im Kopf oder – falls mit `\headlogo` ein Zweitlogo angegeben wurde – im Fuß gesetzt.

Alternativ dazu können die Optionen `ddchead` beziehungsweise `ddcfoot` genutzt werden, welche das Logo explizit entweder im Kopf oder Fuß setzen. Ein mit `\headlogo` angegebenes Zweitlogo wird durch `ddchead=<Einstellung>` definitiv unterdrückt, `ddcfoot=<Einstellung>` setzt das DRESDEN-concept-Logo in jedem Fall in den Seitenfuß. Die Verwendung einer der drei Optionen führt folglich zur Deaktivierung der anderen beiden.

Das DRESDEN-concept-Logo wird standardmäßig sowohl im Kopf als auch im Fuß in der gleichen Höhe gesetzt, wie das Logo der Technischen Universität Dresden und kann zumindest für den Kopf nicht geändert. Wird es im Fuß verwendet, lässt sich die Größe über die Option `footlogoheight` ändern. Sollte nach einer Vergrößerung der Darstellung die Höhe des Fußbereiches nicht ausreichen, so kann diese über `extrabottommargin=<Höhe>` respektive `cdfoot=<Höhe>` angepasst werden. Die möglichen Werte für die drei Schlüssel sind:

`ddc=false/off/no`

Bei den `tudheadings`-Seitenstile erscheint kein Logo von DRESDEN-concept.

`ddc=true/on/yes` (Säumniswert)

Das Logo von DRESDEN-concept wird im Kopf beziehungsweise im Fuß verwendet. Die Wahl der Farbe des Logos geschieht passend zur farblichen Ausprägung der Seite selbst.

Die Farbe des DRESDEN-concept-Logos wird in Abhängigkeit der farblichen Ausprägung des

Layouts (Option `cd=<Einstellung>`) automatisch gewählt. Dies lässt sich manuell ändern:

`ddc=color`

Im Kopf oder Fuß wird die achtfarbige 4C-Variante des DRESDEN-concept-Logos genutzt.

`ddc=colorblack`

Es wird das achtfarbige Logo mit schwarzem DRESDEN-concept-Schriftzug anstelle des grauen verwendet. Für den Fuß wird der grüne Claim durch einen schwarzen ersetzt. Dies ist insbesondere für kleine Darstellungen des Logos im Fuß sinnvoll.

`ddc=gray/grey`

Dies Ausgabe des DRESDEN-concept-Logos erfolgt in Graustufen.

`ddc=black`

Verwendung des Logos in Graustufen mit schwarzem Schriftzug.

`ddc=blue`

Schriftzug und Logo werden in Abstufungen der Hausfarbe HKS41 gesetzt.

`ddc=white`

Das DRESDEN-concept-Logo sowie der dazugehörige Schriftzug sind vollständig weiß.

v2.02  
v2.03

```
\begin{tudpage}[<Sprache> || <Parameterliste>]
. . .
\end{tudpage}
language=<Sprache> (Parameter)
columns=<Anzahl> (Parameter)
pagestyle=<Einstellung> (Parameter)
cdfont=<Einstellung> (Parameter)
cdhead=<Einstellung> (Parameter)
cdfoot=<Einstellung> (Parameter)
headlogo=<Dateiname> (Parameter)
footlogo=<Dateinamenliste> (Parameter)
ddc=<Einstellung> (Parameter)
ddchead=<Einstellung> (Parameter)
ddcfoot=<Einstellung> (Parameter)
```

(siehe `cdfont` auf Seite 16)

(siehe `cdhead` auf Seite 27)

(siehe `cdfoot` auf Seite 29)

(siehe `\headlogo` auf Seite 30)

(siehe `\footlogo` auf Seite 30)

(siehe `ddc` auf Seite 31)

(siehe `ddchead` auf Seite 31)

(siehe `ddcfoot` auf Seite 31)

Die `tudpage`-Umgebung hat ihren Ursprung in einer früheren Version, als die `tudheadings`-Seitenstile noch nicht verfügbar waren, welche mittlerweile anstelle dieser Umgebung verwendet werden können. Für die `tudpage`-Umgebung lassen sich verschiedene Parameter als optionales Argument angeben. Wird das Paket **babel** genutzt, kann die genutzte Sprache mit `language=<Sprache>` geändert werden, was zur Anpassung der sprachspezifischen Trennungsmuster und Bezeichner führt. Wurde das Paket **multicol** geladen, wird mit dem Parameter `columns=<Anzahl>` der Inhalt der Umgebung mehrspaltig gesetzt. Mit `pagestyle` kann der Seitenstil angepasst werden, wobei `headings`, `plain` und `empty` gültige Werte sind.

Die weiteren Parameter entsprechen in ihrem Verhalten den gleichnamigen Klassenoptionen oder Befehlen, wirken sich jedoch nur innerhalb der `tudpage`-Umgebung aus. Das Verhalten sowie gültige Wertzuweisungen ist auf den angegebenen Seiten dokumentiert.

## 2.2.4 Der Titel und die Umschlagseite

Für das Erstellen eines Titels mit dem Befehl `\maketitle` wird mit der KOMA-Script-Option `titlepage=<Ein-Aus-Wert>` festgelegt, ob dieser in Gestalt einer ganzen Titelseite oder nur als



Titelkopf erscheinen soll. Für den Titel im Corporate Design der Technischen Universität Dresden werden alle Felder unterstützt, welche bereits durch KOMA-Script definiert sind. Darüber hinaus werden für die TUD-Script-Klassen weitere Felder bereitgestellt, welche Auswirkungen auf die Gestalt des Titels haben. Diese werden nachfolgend in diesem Abschnitt erläutert. Der Titel – bestehend aus einem möglichen Schmutztitel (`\extratitle`) inklusive dazugehöriger Rückseite (`\frontispiece{<Frontispiz>}`), der eigentlichen Titelseite respektive des Titelkopfes und der nachgelagerten Elementen – kann mit `\maketitle` ausgegeben werden. Außerdem kann im zweispaltigen Satz `\maketitleonecolumn` verwendet werden, womit eine einspaltige Ergänzung nach dem Titel selbst ermöglicht wird.

Zusätzlich zum Titel lässt sich mit `\makecover` eine Umschlagseite erzeugen. Diese kann insbesondere für gebundene Arbeiten verwendet werden. Es wird – im Vergleich zum Titel – lediglich einer reduzierte Anzahl an Feldern auf dieser ausgegeben.

v2.01	<code>\maketitle[&lt;Seitenzahl&gt;    &lt;Parameterliste&gt;]</code>	
v2.02	<code>pagenumber=&lt;Seitenzahl&gt;</code> (Parameter)	
v2.03	<code>cdgeometry=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>cdgeometry</code> auf Seite 22)
v2.06	<code>cdfont=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>cdfont</code> auf Seite 16)
	<code>cdhead=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>cdhead</code> auf Seite 27)
	<code>cdfoot=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>cdfoot</code> auf Seite 29)
	<code>headlogo=&lt;Dateiname&gt;</code> (Parameter)	(siehe <code>\headlogo</code> auf Seite 30)
	<code>footlogo=&lt;Dateinamenliste&gt;</code> (Parameter)	(siehe <code>\footlogo</code> auf Seite 30)
	<code>ddc=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>ddc</code> auf Seite 31)
	<code>ddchead=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>ddchead</code> auf Seite 31)
	<code>ddcfoot=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>ddcfoot</code> auf Seite 31)

Der Befehl `\maketitle` setzt für `cdtitle=false` den normalen KOMA-Script-Titel, ansonsten wird die Titelseite im Corporate Design der Technischen Universität Dresden erzeugt. Die letztere Variante ist im Vergleich zum Standardtitel um eine Vielzahl von Feldern erweitert worden und erlaubt insbesondere die Angabe von Daten für das Deckblatt einer akademischen Abschlussarbeit. Die einzelnen Felder werden später in diesem Abschnitt erläutert. Wird das Dokument doppelseitig und mit rechts öffnenden Kapiteln gesetzt,<sup>25</sup> so wird zusätzlich die Option `cleardoublespecialpage` einbezogen. Dies ist insbesondere bei den Befehlen `\uppertitleback` respektive `\lowertitleback` für die Titelseite zu beachten.

Das optionale Argument erlaubt – ebenso wie bei den KOMA-Script-Klassen – die Änderung der Seitenzahl der Titelseite. Diese wird jedoch nicht ausgegeben, sondern beeinflusst lediglich die Zählung. Sie sollten hier unbedingt eine ungerade Zahl wählen, da sonst die gesamte Zählung durcheinander gerät. Wird eine Titelseite (KOMA-Script-Option `titlepage=true`) im Corporate Design der Technischen Universität Dresden gesetzt (`cdtitle=true`), können auch die weiterhin aufgeführten Parameter im optionalen Argument verwendet werden. Diese entsprechen in ihrem Verhalten den gleichnamigen Optionen respektive Befehlen, wirken sich jedoch nur lokal und einzig auf die Titelseite aus. So kann beispielsweise die Nutzung eines DRESDEN-concept-Logos auf den Titel beschränkt bleiben.

`\maketitleonecolumn[<Seitenzahl> || <Parameterliste>]{<Einspaltentext>}`

Im zweispaltigen Satz (Klassenoption `twocolumn`) wird mit `\maketitle` die Titelseite selbst immer einspaltig gesetzt. Direkt nach dem Titel folgt normalerweise der zweispaltige Fließtext. Mit dem Befehl `\maketitleonecolumn` kann nach dem Titel zusätzlich weiterer Inhalt – zum Beispiel eine Zusammenfassung respektive eine Kurzfassung – einspaltig gesetzt werden.

<sup>25</sup>KOMA-Script-Optionen `twoside=true` und `open=right`, Standard für `tudscrbook`

Bei einer aktivierten Titelseite (KOMA-Script-Option `titlepage=true`) erfolgt die Ausgabe des Argumentes `{<Einspaltentext>}` direkt nach dieser auf einer oder gegebenenfalls mehreren neuen Seiten ebenfalls einspaltig. Wird der Befehl `\maketitleonecolumn` statt mit einer Titelseite jedoch mit einem Titelkopf (KOMA-Script-Option `titlepage=false`) zum Einsatz, so folgt diesem die einspaltige Textpassage aus dem obligatorischen Argument direkt, wobei gegebenenfalls bei entsprechendem Inhalt ein automatischer Seitenumbruch erfolgt. Danach wird direkt und ohne zusätzlichen Umbruch auf das zweispaltige Layout umgeschaltet.

Der optionale Parameter von `\maketitleonecolumn` kann äquivalent zu `\maketitle` für die Änderung der Seitenzahl, der verwendeten Schrift sowie zur Anpassung von Kopf und Fuß verwendet werden. Dabei ist zu beachten, dass ein Großteil der Parameter nur Auswirkungen haben, falls eine Titelseite (KOMA-Script-Option `titlepage=true`) verwendet wird.

v2.02

v2.03

```
\makecover[<Seitenzahl> || <Parameterliste>]
```

```
pagenumber=<Seitenzahl> (Parameter)
```

```
cdgeometry=<Ein-Aus-Wert> (Parameter)
```

```
cdfont=<Einstellung> (Parameter)
```

(siehe `cdfont` auf Seite 16)

```
cdhead=<Einstellung> (Parameter)
```

(siehe `cdhead` auf Seite 27)

```
cdfoot=<Einstellung> (Parameter)
```

(siehe `cdfoot` auf Seite 29)

```
headlogo=<Dateiname> (Parameter)
```

(siehe `\headlogo` auf Seite 30)

```
footlogo=<Dateinamenliste> (Parameter)
```

(siehe `\footlogo` auf Seite 30)

```
ddc=<Einstellung> (Parameter)
```

(siehe `ddc` auf Seite 31)

```
ddchead=<Einstellung> (Parameter)
```

(siehe `ddchead` auf Seite 31)

```
ddcfoot=<Einstellung> (Parameter)
```

(siehe `ddcfoot` auf Seite 31)

Eine Umschlagseite wird zumeist für gebundene Abschlussarbeiten verlangt, um diese beispielsweise für einen Prägedruck auf dem Buchdeckel zu verwenden. Deshalb ist die farbige Ausprägung der Umschlagseite auch deaktiviert, wenn diese für das restliche Dokument aktiv ist (`cd=color`). Dies kann jedoch jederzeit mit `cdcover=<Einstellung>` überschrieben werden.

Wird `cdcover=true` gewählt, so wird die Umschlagseite im Corporate Design der Technischen Universität Dresden gesetzt. Auf dieser werden der Titel des Dokumentes, die Typisierung durch `\thesis` und/oder `\subject` sowie der Autor oder respektive die Autoren und gegebenenfalls der mit `\publishers` angegebene Verlag ausgegeben. Für die Einstellung `cdcover=false` wird lediglich der normale KOMA-Script-Titel als separate Umschlagseite ausgegeben.

v2.02

Die Titelseite selbst gehört immer zum Buchblock und sollte daher im gleichen Satzspiegel gesetzt werden. Dem entgegen steht die Umschlagseite, welche zumeist in einem anderen Layout erscheint. Normalerweise wird das Cover – unabhängig von der Option `cdgeometry` – im asymmetrischen Satzspiegel des Corporate Designs gesetzt. Mit `cdgeometry=false` im optionalen Argument kann das Verhalten geändert werden. In diesem Fall erscheint auch die Umschlagseite im Buchblock des restlichen Dokumentes. Allerdings können für diese Einstellung die Seitenränder mit den Befehlen `\coverpagetopmargin`, `\coverpageleftmargin`, `\coverpagerightmargin` sowie `\coverpagebottommargin` durch den Nutzer frei angepasst werden. Mehr dazu ist im KOMA-Script-Handbuch<sup>26</sup> zu finden.

Außerdem kann mit dem optionalen Argument die Seitenzahl der Umschlagseite geändert werden. Diese wird jedoch nicht ausgegeben, sondern beeinflusst lediglich die Zählung. Sie sollten hier unbedingt eine ungerade Zahl wählen, da sonst die gesamte Zählung durcheinander gerät. Die weiterhin aufgeführten Parameter entsprechen in ihrem Verhalten beziehungsweise ihrer Funktion den gleichnamigen Optionen respektive Befehlen, wirken sich jedoch nur lokal

<sup>26</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

und einzig auf die Umschlagseite aus.

v2.01

```
\title{<Titel>}
\subtitle{<Untertitel>}
```

Die Befehle `\title` und `\subtitle` sind selbsterklärend. Anzumerken ist, dass die Schriftstärke des Titels von der Option `headings` abhängt und der Untertitel immer im fetten Schnitt erscheint.

v2.02

v2.06

```
titlepage (Schriftelement)
thesis (Schriftelement)
\raggedtitle
```

Die TUD-Script-Klassen definieren diese neuen Schriftelemente. Dabei wird `titlepage` auf der Titelseite für alle Felder verwendet, welche kein spezielles Schriftelement verwenden, welches ohnehin durch KOMA-Script bereitgestellt wird. Das mit `\thesis` angegebene Feld, in welchem der Typ einer Abschlussarbeit angegeben wird, nutzt das Schriftelement `thesis`.

v2.02

Für alle Felder des Titels und der Umschlagseite lassen sich die verwendeten Schriften anpassen. In Abschnitt 2.2.8 lässt sich nachlesen, wie dies genau funktioniert. Dabei werden für Titel und Umschlagseite sowohl die bereits durch KOMA-Script bereitgestellten Schriftelemente `titlehead`, `subject`, `title`, `subtitle`, `author`, `publishers`, `date` und `dedication` als auch die neuen `titlepage` und `thesis` unterstützt. Der Befehl `\raggedtitle` definiert die Ausrichtung des Titels, welcher standardmäßig linksbündig gesetzt wird.

v2.06

**Beispiel:** Um die Einträge auf der Titelseite zentriert auszugeben, genügt folgende Definition:

```
\let\raggedtitle\centering
```

v2.02

v2.05

v2.06

```
\author{<Autor(en)>}
\authormore{<Autorenzusatz>}
\emailaddress[<Einstellungen>]{<E-Mail-Adresse>}
\emailaddress*{<E-Mail-Adresse>}
\dateofbirth{<Geburtsdatum>}
\placeofbirth{<Geburtsort>}
\matriculationnumber{<Matrikelnummer>}
\matriculationyear{<Immatrikulationsjahr>}
\course{<Studiengang>}
\discipline{<Studienrichtung>}
```

Mit dem Befehl `\author` wird der Autor angegeben. Innerhalb des Argumentes können auch mehrere Autoren aufgeführt werden, wobei diese in diesem Fall jeweils mit `\and` zu trennen sind. Alle weiteren hier vorgestellten Befehle können selbst im Argument von `\author` verwendet werden, wodurch für jeden Autor individuelle Angaben möglich sind.

Mit `\authormore` wird unter dem Autor eine Zeile ausgegeben, welche durch den Anwender frei belegt werden kann. Mit `\emailaddress` kann für jeden Autor eine E-Mail-Adresse angegeben werden, welche als Hyperlink definiert wird, falls das Paket **hyperref** geladen wurde. Das optionale Argument wird an `\hypersetup` aus besagtem Paket übergeben und kann somit für zusätzliche Einstellungen genutzt werden. Mit der Sternversion `\emailaddress*` erfolgt keine Formatierung des Eintrags im Argument.

Sollte das Paket **isodate** oder **datettime2** geladen sein, wird die damit eingestellte Formatierung des Datums durch `\dateofbirth` – wie übrigens bei jedem anderem Datumsfeld der TUD-Script-Klassen auch – für das Geburtsdatum auf dem Titel verwendet. Hierfür wird

entweder `\printdate` von **isodate** oder `\DTMDate` aus **datetime2** genutzt. Mit dem Befehl `\placeofbirth` lässt sich zusätzlich ein Geburtsort angeben.

Die weiteren Befehle als zusätzliche Angabe erklären sich quasi von selbst. Anzumerken ist, dass die mit den Befehlen `\matriculationnumber`, `\matriculationyear`, `\course` sowie `\discipline` gemachten Angaben ebenfalls vom Paket **tudscrsupervisor** innerhalb der task-Umgebung genutzt werden, falls diese denn zum Einsatz kommt.

`\and`

Dieser Befehl wird sowohl bei den L<sup>A</sup>T<sub>E</sub>X-Standardklassen als auch bei den KOMA-Script-Klassen lediglich auf der Titelseite dazu verwendet, mehrere Autoren im Argument von `\author` voneinander zu trennen.

Bei den TUD-Script-Klassen hingegen ist dieser Befehl derart in seiner Funktion erweitert worden, dass damit die Angabe einer kollaborativen Autorenschaft für Abschlussarbeiten innerhalb des Befehls `\author` möglich ist. Außerdem kann er noch im Argument von `\supervisor`, `\referee` sowie `\advisor` verwendet werden, um mehrere Betreuer beziehungsweise Gutachter und Fachreferenten anzugeben. Er ist dabei nicht auf die Verwendung für den Titel allein beschränkt sondern kann auch bei der Angabe von Personen in den entsprechenden Feldern der Umgebungen `task`, `evaluation` und `notice` aus dem Paket **tudscrsupervisor** eingesetzt werden.

**Beispiel:** Angenommen, es soll eine Abschlussarbeit von zwei unterschiedlichen Autoren in kollaborativer Gemeinschaft erstellt werden, so lässt sich die Autorenangaben folgendermaßen gestalten:

```
\author{%
  Mickey Mouse%
  \matriculationnumber{12345678}%
  \dateofbirth{2.1.1990}%
  \placeofbirth{Dresden}%
\and%
  Donald Duck%
  \matriculationnumber{87654321}%
  \dateofbirth{1.2.1990}%
  \placeofbirth{Berlin}%
}
\matriculationyear{2010}
```

Alle zusätzlichen Angaben außerhalb des Argumentes von `\author` werden für beide Autoren gleichermaßen übernommen. Angaben innerhalb des Argumentes von `\author` werden den jeweiligen, mit `\and` getrennten Autoren zugeordnet. Mehr dazu ist im Minimalbeispiel in Unterkapitel 6.3.

v2.05

```
\date[<Suffix> || <Parameterliste>]{<Datum>}
\date* [<Suffix> || <Parameterliste>]{<Datum>}
before=<Präfix> (Parameter)
after=<Suffix> (Parameter)
place=<Ort> (Parameter)
\defensedate{<Verteidigungsdatum>}
```

v2.05

Mit dem Befehl `\date` lässt sich das Datum angeben. Über das optionale Argument können die beiden Parameter `before` und `after` genutzt werden, um ergänzende Angaben vor beziehungsweise nach dem eigentlichen Datum auszugeben. Die Sternversion `\date*` setzt den

mit `\place` angegebenen Ort vor das Datum. Dies geschieht auch für die normale Version von `\date`, wenn der Parameter `place` verwendet wird.

Das Datum wird bei normalen Dokumenten direkt nach dem Autor respektive den Autoren ausgegeben. Bei Abschlussarbeiten – aktiviert durch die Verwendung von `\thesis` oder `subjectthesis` in Verbindung mit `\subject` – erscheint dieses am Ende der Titelseite als Abgabedatum. Außerdem kann in diesem Fall mit dem Befehl `\defensedate` das Datum der Verteidigung angegeben werden, wie es beispielsweise bei dem Druck von Dissertationen üblich ist.

Sollte eines der Pakete `isodate` oder `datetime2` geladen sein, so wird mit `\printdate` beziehungsweise `\DTMDate` die durch das jeweilige Paket eingestellte Ausgabeformatierung des Datums für alle Datumsfelder des Dokumentes und folglich auch für die beiden Felder `\date` und `\defensedate` verwendet.

```
\thesis{<Typisierung>}
\subject{<Typisierung>}
```

Mit diesen beiden Befehlen kann der Typ der Dokumentes beziehungsweise der Abschlussarbeit angegeben werden. Während der Befehl `\thesis` den Inhalt des Feldes unter dem Titel vertikal zentriert auf der Titelseite ausgibt, erscheint der Inhalt des Befehls `\subject` oberhalb des Titels. Es können auch beide Befehle parallel mit unterschiedlichen Inhalten verwendet werden. Der Befehl `\thesis` dient den TUD-Script-Dokumentklassen außerdem zur Erkennung von Abschlussarbeiten gedacht, da für diese spezielle Felder bereitgehalten werden und auch die Titelseite leicht geändert gesetzt wird.

Des Weiteren ist es bei beiden Befehlen möglich, spezielle Werte als Argument zur Typisierung des Dokumentes zu verwenden. Diese werden entsprechend der gewählten Dokumentensprache – entweder Deutsch oder Englisch – entschlüsselt und gesetzt. Die möglichen Werte sind Tabelle 2.1 zu entnehmen. Dabei ist zu beachten, dass das Setzen eines speziellen Wertes für *entweder* `\thesis` oder `\subject` möglich ist. Die Verwendung eines der genannten Werte führt immer dazu, dass das Dokument als Abschlussarbeiten erkannt und die erweiterte Titelseite aktiviert wird. Gleichzeitig wird damit die Option `subjectthesis` beeinflusst. Sollte vom Anwender kein explizites Verhalten für `subjectthesis` definiert sein, so führt die Verwendung von `\thesis{<Wert>}` zu `subjectthesis=false` und `\subject{<Wert>}` zu `subjectthesis=true`.

```
subjectthesis=<Ein-Aus-Wert> Voreinstellung: false | \subject{<Tabelle 2.1>}: true
```

Der Befehl `\thesis` dient den TUD-Script-Hauptklassen zur Unterscheidung zwei unterschiedlicher Ausprägungen der Titelseite und ist speziell für Abschlussarbeiten gedacht. Außerdem kann bei der Nutzung spezieller Werte aus Tabelle 2.1 innerhalb des Argumentes von `\subject` ebenfalls das Verhalten für Abschlussarbeiten aktiviert werden, wobei hierdurch die Einstellung `subjectthesis=true` automatisch vorgenommen wird.

Für den Standardfall – bekanntlich `subjectthesis=false` – wird der durch `\thesis` gegebene Typ der Abschlussarbeit sowie der gegebenenfalls durch `\graduation` gesetzte angestrebte Abschluss in großen Lettern und sehr zentral auf der Titelseite gesetzt. Die Verwendung von `\subject` ist hierbei weiterhin möglich. Wird die Option mit `subjectthesis=true` aktiviert, so wird die mit `\thesis` gesetzte Bezeichnung nicht unterhalb sondern oberhalb des Titels an der Stelle von `\subject` ausgegeben. Der mit `\graduation` angegebene Abschluss wird weiterhin unter dem Titel, allerdings in schlankerer Schrift gesetzt. Eine etwaige Verwendung des Befehls `\subject` wird in diesem Fall ignoriert.

Tabelle 2.1 Spezielle Werte zur Typisierung des Dokumentes für `\thesis` und `\subject`

Wert	Bezeichner	Deutsch	Englisch
habil	<code>\habilitationname</code>	Habilitation	Habilitation
diss	<code>\dissertationname</code>	Dissertation	Dissertation
phd	<code>\dissertationname</code>	Dissertation	Dissertation
diploma	<code>\diplomathesisname</code>	Diplomarbeit	Diploma Thesis
master	<code>\masterthesisname</code>	Master-Arbeit	Master Thesis
bachelor	<code>\bachelorthesisname</code>	Bachelor-Arbeit	Bachelor Thesis
student	<code>\studentthesisname</code>	Studienarbeit	Student Thesis
evidence	<code>\studentresearchname</code>	Großer Beleg	Student Research Project
project	<code>\projectpapername</code>	Projektarbeit	Project Paper
seminar	<code>\seminarpapername</code>	Seminararbeit	Seminar Paper
term	<code>\termpapername</code>	Hausarbeit	Term Paper
research	<code>\researchname</code>	Forschungsbericht	Research Report
log	<code>\logname</code>	Protokoll	Log
report	<code>\reportname</code>	Bericht	Report
internship	<code>\internshipname</code>	Praktikumsbericht	Internship Report

`subjectthesis=false/off/no`

Die Ausgabe des Typs der Abschlussarbeit (`\thesis`) selbst sowie des angestrebten Abschlusses (`\graduation`) erfolgt in großen Lettern zentral auf der Titelseite.

`subjectthesis=true/on/yes` (Säumniswert)

Der Typ der Abschlussarbeit (`\thesis`) wird oberhalb des Titels in der Betreffzeile gesetzt. Der angestrebte Abschluss (`\graduation`) wird zentral ausgegeben.

v2.02

`\graduation[<Kurzform>]{<Grad>}`

Mit diesem Befehl wird der angestrebte akademische Grad auf der Titelseite ausgegeben. Da dies nur mit einer Abschlussarbeit erreicht werden kann erfolgt die Ausgabe nur, wenn entweder `\thesis` oder `\subject` verwendet wurde, wobei bei letzterem Befehl im Argument zwingend ein Wert aus Tabelle 2.1 verwendet werden muss.

Bei der Ausgabe des akademischen Grades hat die Option `subjectthesis` Einfluss auf die Ausgabe auf der Titelseite. Bei `subjectthesis=false` wird der Abschluss – ähnlich wie der Typ der Abschlussarbeit – zentral und in relativ großen Lettern gesetzt. Für `subjectthesis=true` erfolgt die Ausgabe kleiner und in weniger starken Buchstaben.

```
\supervisor{<Name(n)>}
\referee{<Name(n)>}
\advisor{<Name(n)>}
\professor{<Name>}
```

Mit `\supervisor`, `\referee` und `\advisor` werden die Betreuer einer Abschlussarbeit beziehungsweise die Gutachter und Fachreferenten einer Dissertation angegeben. Zusätzlich kann mit `\professor` der betreuende Hochschullehrer beziehungsweise die betreuenden Professoren für studentische Arbeiten angegeben werden. Die Angabe mehrerer Person erfolgt wie beim Befehl `\author` durch die Trennung mittels `\and`.

v2.06

`titlesignature=<Ein-Aus-Wert>` Voreinstellung: `false`

Einige Lehrstühle an der Technischen Universität Dresden verlangen eine Unterschrift des Autors einer Abschlussarbeit direkt auf der Titelseite. Mit dem Aktivieren dieser Option wird ein

solches Feld am unteren Seitenrand des Titels erzeugt.

```
\titledelimiter{<Trennzeichen>}
```

Für den Titel und die Umschlagseite werden durch die TUD-Script-Klassen eine Reihe von zusätzlichen Feldern bereitgestellt. Einigen dieser Felder wird eine Beschreibung (*siehe* Unterkapitel 2.4) vorangestellt. Dazwischen wird bei der Ausgabe ein Trennzeichen eingefügt. Ein Doppelpunkt gefolgt von einem Leerzeichen (`:\nobreakspace`) ist hierfür die Voreinstellung. Mit dem Befehl `\titledelimiter` lässt sich dieses Trennzeichen beliebig an die individuellen Wünsche des Anwenders anpassen.

```
\extratitle{<Schmutztitel>}
\frontispiece{<Frontispiz>}
\titlehead{<Kopf>}
\publishers{<Verlag>}
\thanks{<Fußnote>}
\uppertitleback{<Titelrückseitenkopf>}
\lowertitleback{<Titelrückseitenfuß>}
\dedication{<Widmung>}
```

Diese Befehle entsprechen den in ihrem Verhalten den originalen Pendanten der KOMA-Script-Klassen und sollen hier der Vollständigkeit halber erwähnt werden.

v2.06

Die Ausgabe des mit `\extratitle` definierten Schmutztitels – welcher beliebig gestaltet und formatiert werden kann – und der gegebenenfalls mit `\frontispiece` definierten Rückseite erfolgt als Bestandteil der Titelei mit `\maketitle` vor der eigentlichen Titelseite. Mit dem Befehl `\titlehead` kann ein zusätzlicher, beliebig formatierbarer Text oberhalb der Typisierung und des Titels ausgegeben werden. Da die vertikale Position des Dokumenttitels durch das Corporate Design fest vorgegeben ist, kann es – im Gegensatz zu den KOMA-Script-Klassen – passieren, dass der Kopf des Haupttitels selbst in die Kopfzeile ragt. Dies wird durch die TUD-Script-Klassen nicht geprüft und muss gegebenenfalls vom Anwender kontrolliert werden. Der mit dem Befehl `\publishers` definierte Inhalt muss nicht zwingende einen Verlag bezeichnen sondern kann auch andere Informationen beinhalten, welche am Ende der Titelseite ausgegeben werden sollen.

Fußnoten werden auf dem Titel nicht mit `\footnote`, sondern mit der Anweisung `\thanks` erzeugt. Diese dienen in der Regel für Anmerkungen bei Titel oder den Autoren. Als Fußnotenzeichen werden dabei Symbole statt Zahlen verwendet. Der Befehl `\thanks` kann nur innerhalb des Arguments einer der Anweisungen für die Titelseite wie beispielsweise `\author` oder `\title` verwendet werden.

Im doppelseitigen Druck lässt sich die Rückseite der Haupttitelseite für weitere Angaben nutzen. Sowohl den Titelrückseitenkopf als auch den Titelrückseitenfuß kann der Anwender mit `\uppertitleback` und `\lowertitleback` frei gestalten.

Mit `\dedication` lässt eine separate Widmungsseite zentriert und in etwas größerer Schrift setzen. Die Rückseite ist – wie auch die des Schmutztitels – grundsätzlich leer. Die Widmung wird mit der restlichen Titelei ausgegeben und muss daher vor der Nutzung von `\maketitle` angegeben werden.

## 2.2.5 Die Teileseite

v2.02

Wird für die Teileseiten das Layout des Corporate Designs verwendet, so wird der Seitenstil dieser (`\partpagestyle`) auf `plain.tudheadings` gesetzt. Möchten Sie stattdessen einen ande-

ren Seitenstil nutzen, so kann dieser mit `\renewcommand*{\partpagestyle}{<Seitenstil>}` angepasst werden.

v2.06

`\setpartsubtitle{<Untertitel>}`  
`partsubtitle` (Schriftelement)

Mit `\setpartsubtitle{<Untertitel>}` kann für Teile nach der Überschrift selbst ein Untertitel gesetzt werden. Der Befehl muss in gleicher Weise wie `\setpartpreamble` vor der Verwendung von `\part` oder den davon abgeleiteten Varianten angegeben werden. Mit dem Schriftelement `partsubtitle` lässt sich die Schrift für den gegebenen Untertitel verändern. In Abschnitt 2.2.8 ist zu finden, wie es angepasst werden kann.

`parttitle=<Ein-Aus-Wert>` Voreinstellung: false

Diese Option ermöglicht es, den mit `\title` gegebenen Titel des Dokumentes selbst in großer Schrift auf einer Teileseite auszugeben, die Bezeichnung des mit `\part{<Bezeichnung>}` erzeugten Teils wird in diesem Fall als Untertitel direkt darunter gesetzt. Diese Layout-Variante findet sich im Handbuch für das Corporate Design der Technischen Universität Dresden. Für die Klassen **tudscrtcl** sowie **tudscrposter** ist diese Option nicht verfügbar.

`parttitle=false/off/no`

Die Bezeichnung des Teils erscheint in großer Schrift, der Titel des Dokumentes wird nicht ausgegeben.

`parttitle=true/on/yes` (Säumniswert)

Der angegebene Titel wird in großer Auszeichnung auf der Teileseite gesetzt, die Bezeichnung des Teils selber als Untertitel.

## 2.2.6 Die Kapitelseite

`chapterpage=<Ein-Aus-Wert>` Voreinstellung: false | cd=color: true

Mit dieser Einstellung kann die Überschrift eines Kapitels separat auf einer Seite ausgegeben werden. Der nachfolgende Text wird auf der nächsten beziehungsweise bei doppelseitigem Satz und rechts öffnenden Kapiteln<sup>27</sup> auf der übernächsten Seite ausgegeben. Die in diesem Fall erzeugte Rückseite wird in ihrer Ausprägung – wie auch Teileseiten – durch die Einstellung von `cleardoublespecialpage` bestimmt. Beim farbigen Layout ist diese Option standardmäßig aktiviert. Für die Klassen **tudscrtcl** sowie **tudscrposter** ist diese Option nicht verfügbar.

`chapterpage=false/off/no`

Es gibt keine Sonderstellung von Kapiteln, der nachfolgende Text wird direkt unter der Überschrift respektive nach der mit `\setchapterpreamble` erzeugten Kapitelpräambel auf der gleichen Seite ausgegeben.

`chapterpage=true/on/yes` (Säumniswert)

Die Kapitelüberschrift und gegebenenfalls die Kapitelpräambel werden auf einer separaten Seite gesetzt. Der folgende Text erscheint auf der nächsten respektive übernächsten Seite, *siehe auch* `cleardoublespecialpage`.

v2.02

Mit `\renewcommand*{\chapterpagestyle}{<Seitenstil>}` lässt sich übrigens – unabhängig von der Option `chapterpage` – der Seitenstil von Kapiteln anpassen. Bei der Verwendung von Kapitelseiten mit `chapterpage=true` ist außerdem das Aktivieren der KOMA-Script-Option `chapterprefix=<Ein-Aus-Wert>` empfehlenswert. Damit werden die Kapitelüberschriften mit

<sup>27</sup>KOMA-Script-Optionen `twoside=true` und `open=right`, Standard für **tudscrbook**



einer Vorsatzzeile gesetzt. Wird ein nummeriertes Kapitel erzeugt, so wird zunächst in einer Zeile „Kapitel“ gefolgt von der aktuellen Kapitelnummer ausgegeben, in der nächsten Zeile wird anschließend die eigentliche Überschrift in linksbündigem Flattersatz ausgegeben. Mehr dazu ist der KOMA-Script-Dokumentation<sup>28</sup> zu entnehmen.

v2.06

```
\setchaptersubtitle{<Untertitel>}
chaptersubtitle (Schriftelement)
```

Mit `\setchaptersubtitle{<Untertitel>}` kann für Kapitel nach der Überschrift selbst ein Untertitel gesetzt werden. Der Befehl muss in gleicher Weise wie `\setchapterpreamble` vor der Verwendung von `\chapter` oder den davon abgeleiteten Varianten angegeben werden. Mit dem Schriftelement `chaptersubtitle` lässt sich die Schrift für den gegebenen Untertitel verändern. In Abschnitt 2.2.8 ist zu finden, wie es angepasst werden kann.

## 2.2.7 Vakatsseiten

Automatisch erzeugte Vakatsseiten – auch absichtliche Leerseiten genannt – sind in Abhängigkeit der KOMA-Script-Optionen `twoside=<Einstellung>` und `open=<Methode>` am Beginn von Teilen und Kapiteln in Dokumenten zu finden.<sup>29</sup> Für die Vakatsseiten kann der Seitenstil mit der KOMA-Script-Option `cleardoublepage=<Einstellung>` eingestellt werden.

v2.06

```
cleardoublespecialpage=<Einstellung>    Voreinstellung: true
```

Diese Option wirkt sich lediglich bei aktiviertem doppelseitigem Satz und ausschließlich rechts eröffnenden Seiten für Teile beziehungsweise Kapitel aus.<sup>30</sup> In diesem Fall kann der Stil der darauffolgenden, linken Seite – sprich der Rückseite – beeinflusst werden. Das Normalverhalten sieht vor, dass nach einem Teil die nachfolgende Rückseite unabhängig von der Einstellung für `cleardoublepage` immer als vollständig leere Seite ohne Kopf- oder Fußzeilen gesetzt wird.

Diese Option erlaubt es, das Normalverhalten zu deaktivieren und für die Seite nach der Teilleseite – und abhängig von `chapterpage` auch nach einem Kapitelanfang auf einer separaten Seite – den Seitenstil der Option `cleardoublepage` zu übernehmen. Des Weiteren kann auch ein anderer, bereits definierter Seitenstil gewählt werden. Außerdem kann im farbigen Layout die Rückseite in der gleichen Farbe wie die Vorderseite von Titel, Teil oder Kapitel gesetzt werden. Für die Klassen `tudscrartcl` sowie `tudscrposter` ist diese Option nicht verfügbar.

```
cleardoublespecialpage=false/off/no
```

Die Rückseiten sind vollständig leere Seiten, unabhängig von Option `cleardoublepage`.

```
cleardoublespecialpage=true/on/yes (Säumniswert)
```

Der Seitenstil der Rückseite von Teilen und gegebenenfalls Kapiteln entspricht der Einstellung von `cleardoublepage` für Vakatsseiten.

```
cleardoublespecialpage=current
```

Für die erzeugte Rückseite wird der aktuell definierte Seitenstil (`\pagestyle`) verwendet.

```
cleardoublespecialpage=<Seitenstil>
```

Mit der Angabe von `cleardoublespecialpage=<Seitenstil>` kann ein beliebiger, bereits definierter Seitenstil für die Rückseite nach Teilen und Kapiteln verwendet werden.

```
cleardoublespecialpage=color
```

Im farbigen Layout ist auch die Rückseite von Teilen und Kapiteln farbig, *siehe* `cd`. Die

<sup>28</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

<sup>29</sup>Voreinstellungen `tudscrbook`: `twoside=true`, `open=right`; `tudscrartcl`, `tudscrreprt`: `open=any`, `twoside=false`

<sup>30</sup>KOMA-Script-Option `twoside=true` und `open=right`

Einstellung wirkt sich ebenfalls auf die Rückseite des Titels aus.<sup>31</sup>  
`cleardoublespecialpage=nocolor`

Es werden weiße Rückseiten bei Titel, Teilen und gegebenenfalls Kapiteln erzeugt.

## 2.2.8 Verwendung von Schriftelementen

Vom TUD-Script-Bundle werden weitere Schriftelemente – in Ergänzung zu den bereits durch KOMA-Script bereitgestellten – definiert. Dies sind `titlepage`, `thesis`, `tudheadings` sowie `partsubtitle` und `chaptersubtitle`. Sowohl die durch KOMA-Script definierten als auch alle hier genannten und folgend erläuterten Schriftelemente lassen sich im Bedarfsfall über den Befehl `\addtokomafont{<Schriftelement>}{<Einstellungen>}` anpassen. Mehr dazu ist im KOMA-Script-Handbuch<sup>32</sup> innerhalb des Abschnitts *Textauszeichnungen* zu finden.

## 2.2.9 Verwendung von Feldinhalten

v2.06

```
\getfield{<Feldname>}
```

Mit diesem Makro kann auf den Inhalt eines zuvor angegebenen Feldes zugegriffen werden.

**Beispiel:** Das für dieses Dokument angegebene Datum lautet „9. August 2022“. Es kann wie folgt ausgegeben werden:

```
\getfield{date}
```

Weitere mögliche Argumente sind beispielsweise `faculty`, `department`, `institute` und `chair` sowie `title`, `author`, `thesis`, `professor`, `supervisor` oder `place`.

## 2.2.10 Die Farben des Corporate Designs

Zur Verwendung der Farben des Corporate Designs wird das Paket `tudscrcolor` genutzt. Falls dieses nicht in der Präambel geladen wird – um beispielsweise zusätzliche Optionen aufzurufen – binden die TUD-Script-Klassen dieses automatisch ein. Detaillierte Informationen sind in der Dokumentation von `tudscrcolor` in Unterkapitel 5.1 auf Seite 71 zu finden.

## 2.3 Zusätzliche Optionen und Erweiterungen

Neben den Befehlen für die Anpassung des Layouts an das Corporate Design der Technischen Universität Dresden stellen die TUD-Script-Klassen weitere Befehle und Umgebungen zur Verfügung, um die Anwendung insbesondere für wissenschaftliche Arbeiten zu erleichtern.

### 2.3.1 Zusammenfassung/Kurzfassung

v2.02

v2.04

```
abstract=<Einstellung>
```

Diese Option wird bereits durch KOMA-Script für die Klassen `scartcl` und `screprt` standardmäßig bereitgestellt. Für die Klasse `scrbook` geschieht dies nicht. Dazu heißt es im Handbuch:

<sup>31</sup>siehe `\uppertitleback` und `\lowertitleback` im KOMA-Script-Handbuch

<sup>32</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

Bei Büchern wird in der Regel eine andere Art der Zusammenfassung verwendet. Dort wird ein entsprechendes Kapitel an den Anfang oder Ende des Werks gesetzt. Oft wird diese Zusammenfassung entweder mit der Einleitung oder einem weiteren Ausblick verknüpft. Daher gibt es bei **scrbook** generell keine `abstract`-Umgebung. Bei Berichten im weiteren Sinne, etwa einer Studien- oder Diplomarbeit, ist ebenfalls eine Zusammenfassung in dieser Form zu empfehlen.

Durch die TUD-Script-Klassen wird die `abstract`-Option erweitert. Neben den Auswahlmöglichkeiten, welche bereits KOMA-Script für die Klassen **tudscartcl** und **tudscpreprt** anbietet, kann die Überschrift für die Zusammenfassung außerdem in Gestalt eines Unterkapitels oder für **tudscpreprt** und **tudscrbook** in der Form eines Kapitels ausgegeben werden.

`abstract=false/off/no` (nur **tudscartcl** und **tudscpreprt**)

Es wird keine Überschrift für die `abstract`-Umgebung ausgegeben.

`abstract=true/on/yes` (Säumniswert) (nur **tudscartcl** und **tudscpreprt**)

Wie bei den KOMA-Script-Klassen wird eine zentrierte Überschrift mit dem Bezeichner `\abstractname` vor der eigentlichen Zusammenfassung gesetzt.

`abstract=section/addsec`

Die Überschrift (`\abstractname`) verwendet den Gliederungsbefehl `\section`.

`abstract=chapter/addchap` (Säumniswert für **tudscrbook**) (nur **tudscpreprt** und **tudscrbook**)

Es wird der Befehl `\chapter` für das Setzen der Überschrift (`\abstractname`) genutzt.

`abstract=heading`

Es wird die höchstmögliche Gliederungsebene verwendet. Für **tudscartcl** entspricht dies `abstract=section`, bei **tudscpreprt** und **tudscrbook** `abstract=chapter`.

Abhängig von der gewählten Gliederungsebene der Überschrift wird das Verhalten für das Setzen eines Eintrages ins Inhaltsverzeichnis festgelegt. Ohne oder mit zentrierter Überschrift wird per Voreinstellung kein Eintrag erzeugt. Wird die Überschrift jedoch in Form einer Gliederungsebene gewählt, so erscheint die Zusammenfassung für gewöhnlich im Inhaltsverzeichnis auf der obersten Ebene. Das voreingestellte Verhalten für die Einträge ins Inhaltsverzeichnis kann jederzeit mit folgenden Werten durch den Anwender überschrieben werden.

`abstract=notoc/nottotoc`

Die Zusammenfassung wird definitiv nicht ins Inhaltsverzeichnis eingetragen.

`abstract=toc/totoc`

Es wird auf der obersten Gliederungsebene der aktuell verwendeten Dokumentklasse (`\chapter` oder `\section`) ein nicht nummerierter Eintrag im Inhaltsverzeichnis für die Zusammenfassung gesetzt.

v2.02

`abstract=tocleveldown/leveldown/totocleveldown`

Der Inhaltsverzeichniseintrag wird eine Stufe unterhalb der obersten Gliederungsebene (`\section` oder `\subsection`) erzeugt.

v2.04

`abstract=tocmultiple/totocmultiple/tocaggregate/totocaggregate`

Es wird ein *einzig* Inhaltsverzeichniseintrag für *alle* Zusammenfassungen erstellt.

v2.02

Außerdem kann das Verhalten für die Kolumnentitel durch den Nutzer beeinflusst werden. Diese werden normalerweise automatisch gesetzt, wenn diese über die KOMA-Script-Option `automark` aktiviert wurden und sind von der aktuellen Gliederungsebene der Überschrift abhängig. Werden jedoch mit Hilfe der KOMA-Script-Option `manuallmark` manuelle Kolumnentitel genutzt, müssen diese normalerweise auch für die Zusammenfassung durch den Anwender gesetzt werden. Mit `abstract=markboth` lässt sich allerdings das Setzen der Kolumnentitel unabhängig davon forcieren.

`abstract=markboth`

Unabhängig von der Verwendung manueller oder automatischer Kolumnentitel werden diese auf rechten sowie linken Seiten mit `\abstractname` gesetzt.

`abstract=nomarkboth`

Die Einstellung für manuelle oder automatische Kolumnentitel werden beachtet und abhängig von der verwendeten Gliederungsebene der Überschrift gesetzt.

Mit dem optionalen Parameter `markboth` der `abstract`-Umgebung kann der Kolumnentitel mit einem beliebigen Inhalt gesetzt werden.

Häufig wird für Abschlussarbeiten verlangt, neben der deutschsprachigen auch noch eine englischsprachige Zusammenfassung zu verfassen. Mit der Einstellung `abstract=multiple` lassen sich mehrere Zusammenfassungen auf einer Seite ausgeben – sofern genügend Platz vorhanden ist. Außerdem kann die standardmäßige vertikale Zentrierung der `abstract`-Umgebung auf einer Seite unterdrückt werden. Diese Einstellungen zur Positionierung der Zusammenfassungen innerhalb der `abstract`-Umgebung werden nur wirksam, wenn eine Titelseite (KOMA-Script-Option `titlepage=true`) und *keine* Überschriften in Form von Kapiteln (`abstract=chapter`) verwendet werden.

`abstract=single/one/simple`

Jede Zusammenfassung wird auf einer eigenen Seite beziehungsweise im zweispaltigen Satz in einer neuen Spalte ausgegeben.

v2.02

`abstract=multiple/multi/all/aggregate`

Zusammenfassungen, welche mit `\nextabstract` getrennt wurden, werden direkt nacheinander auf der gleichen Seite ausgegeben, wenn ausreichend Platz auf dieser vorhanden sein sollte. Ist die Klassenoption `twocolumn` aktiviert, erfolgt die Ausgabe aller Zusammenfassungen ohne Spaltenumbruch.

`abstract=fill/fil/vfil/vfill`

Alle Zusammenfassungen werden bei der Ausgabe auf einer Seite vertikal zentriert. Diese Einstellung steht für den zweispaltigen Satz (Klassenoption `twocolumn`) nicht zur Verfügung.

`abstract=nofill/nofil/novfil/novfill`

Die Ausgabe erfolgt wie im normalen Fließtext auch.

v2.02

```
\begin{abstract}[<Sprache> || <Parameterliste> ]
. . .
\end{abstract}
\nextabstract[<Sprache> || <Parameterliste> ]
language=<Sprache> (Parameter)
markboth=<Einstellung> (Parameter)
pagestyle=<Seitenstil> (Parameter)
columns=<Anzahl> (Parameter)
option=<Einstellung> (Parameter)
```

(siehe `abstract` auf Seite 42)

Die `abstract`-Umgebung dient speziell für die Ausgabe einer Zusammenfassung, entweder zu Beginn eines Dokumentes oder beispielsweise vor einem Teil oder Kapitel. Wird ein Titelpfand und keine Titelseite verwendet (KOMA-Script-Option `titlepage=false`), so wird eine Zusammenfassung – identisch zu den KOMA-Script-Klassen – in einer `quotation`-Umgebung ausgegeben, wobei die Überschrift *nicht* in der Form einer Gliederungsebene gesetzt wird. Diese Umgebung hat jedoch den Nachteil, dass die KOMA-Script-Option `parskip=<Methode>` nicht beachtet wird. Um dies zu beheben, kann das Paket **quoting** geladen werden, wodurch

stattdessen die Umgebung `quoting` verwendet wird.

Mit der zuvor erläuterten Option `abstract` kann eingestellt werden, in welcher Gestalt die Zusammenfassung ausgegeben werden soll. Des Weiteren lässt sich jede `abstract`-Umgebung individuell über weitere Parameter als optionales Argument anpassen. Damit lassen sich gegebenenfalls für eine bestimmte `abstract`-Umgebung die globalen Einstellungen der Option `abstract` lokal ändern und gezielt anpassen.

Wird das Paket **babel** durch den Anwender geladen, kann mit dem optionalen Parameter `language=<Sprache>` die Sprache innerhalb der `abstract`-Umgebung geändert werden. Dafür muss die gewünschte Sprache bereits mit dem Laden von **babel** entweder als Paketoption oder besser noch als Klassenoption angegeben worden sein. Dadurch werden innerhalb der Umgebung die Bezeichnung `\abstractname` und die Trennungsmuster sprachspezifisch angepasst. Die gewünschte Sprache kann auch ohne die Verwendung des Parameters `language` direkt als optionales Argument übergeben werden.

v2.02

Mit `markboth` können die gesetzten Kolumnentitel beeinflusst werden. Wird `markboth=false` angegeben, werden automatische respektive manuelle Kolumnentitel verwendet. Die Einstellung `markboth=true` wiederum setzt diese für linke und rechte Seiten auf `\abstractname`. Zusätzlich lässt sich mit `markboth=<Kolumnentitel>` der Kolumnentitel direkt festlegen. So können die Spalten beispielsweise mit der Verwendung von `markboth={}` auch gelöscht werden. Sollte `markboth` aktiviert werden, so wird in der Umgebung automatisch der Seitenstil `headings` genutzt – falls eine Titelseite (KOMA-Script-Option `titlepage=true`) verwendet wird. Mit dem Parameter `pagestyle` kann dieser auch manuell angegeben werden, die `tudheadings`-Seitenstile werden dabei ebenfalls unterstützt werden.

Wurde das Paket **multicol** geladen, kann mit dem Parameter `columns=<Anzahl>` die Zusammenfassung mehrspaltig gesetzt werden. Dem Parameter `option` können alle gültigen, bereits erläuterten Werte der Option `abstract` übergeben werden. Die damit gemachten Einstellungen wirken sich – im Gegensatz zur Variante der späten Optionenwahl<sup>33</sup> respektive als Angabe einer Klassenoption – lediglich lokal auf die verwendete `abstract`-Umgebung aus.

v2.02

Sollen mehrere Zusammenfassungen erzeugt und dabei die Einstellungen `abstract=single` beziehungsweise `abstract=multiple` sowie `abstract=fill` respektive `abstract=nofill` beachtet werden, so ist die `abstract`-Umgebung nur einmal zu verwenden. Innerhalb dieser müssen die einzelnen Zusammenfassungen jeweils mit `\nextabstract` voneinander getrennt werden. Der Befehl akzeptiert dabei im optionalen Argument alle Parameter, welche auch von der `abstract`-Umgebung selbst unterstützt werden. Das Minimalbeispiel in Unterkapitel 6.2 auf Seite 84 zeigt hierfür das notwendige Vorgehen.

Wird die `abstract`-Umgebung innerhalb des Argumentes der Befehle `\setpartpreamble` beziehungsweise `\setchapterpreamble` verwendet, so wird die Überschrift – im Fall, dass nicht `abstract=false` gewählt ist – *immer* in Textgröße und zentriert gesetzt.

### Umbenennung der Zusammenfassung

Mit dem KOMA-Script-Befehl `\renewcaptionname` kann der Bezeichner – sprich der Wortlaut – der für die `abstract`-Umgebung verwendeten Überschrift verändert werden. Mehr dazu ist in Unterkapitel 2.4 zu finden.

**Beispiel:** Die Überschrift der `abstract`-Umgebung soll für die Sprache `ngerman` von „Zusammenfassung“ in „Kurzfassung“ umbenannt werden. Das Makro `\renewcaptionname` erwartet die drei obligatorischen Argumente `{<Sprache>}{<Makro>}{<Inhalt>}`:

<sup>33</sup>`\TUDoption{abstract}{<Einstellung>}` oder `\TUDoptions{abstract=<Einstellung>}`

`\renewcaptionname{ngerman}{\abstractname}{Kurzfassung}`

## 2.3.2 Selbstständigkeitserklärung und Sperrvermerk

v2.02  
v2.04

`declaration=<Einstellung>` Voreinstellung: true

Mit `declaration` kann äquivalent zur Option `abstract` die Gestaltung von Selbstständigkeitserklärung und Sperrvermerk angepasst werden. Zur Ausgabe der Erklärungen werden die Umgebung `declarations` sowie die Befehle `\declaration` beziehungsweise `\confirmation` und `\blocking` bereitgestellt.

Die beiden Optionen `abstract` und `declaration` ähneln sich sehr stark. Alle möglichen Wertzuweisungen für `declaration` wurden bereits bei der Beschreibung von `abstract` ausführlich erläutert. Deshalb geschieht dies hier in einer etwas kürzeren Ausführung. Sollte Ihnen eine Erläuterung etwas dürftig erscheinen, so hilft mit Sicherheit ein Blick zur Erklärung der Option `abstract` in Abschnitt 2.3.1 auf Seite 42.

Die möglichen Werte für die Gestaltung der Überschrift werden nachfolgend genannt. Im Gegensatz zur Option `abstract` stehen die beiden Einstellungen `declaration=true` und `declaration=false` auch für die Klasse **tudscrbook** zur Verfügung.

`declaration=false/off/no`

Es wird keine Überschrift über den Erklärungen selbst ausgegeben.

`declaration=true/on/yes` (Säumniswert)

Eine zentrierte Überschrift mit dem Bezeichner `\confirmationname` vor der Selbstständigkeitserklärung beziehungsweise `\blockingname` vor dem Sperrvermerk wird gesetzt.

`declaration=section/addsec`

Die Überschrift verwendet den Gliederungsbefehl `\section`.

`declaration=chapter/addchap` (Säumniswert für **tudscrbook**) (nur **tudscrreprt** und **tudscrbook**)

Es wird der Befehl `\chapter` für das Setzen der Überschrift genutzt.

`declaration=heading`

Es wird die höchstmögliche Gliederungsebene verwendet. Für **tudscrartcl** entspricht dies `declaration=section`, bei **tudscrreprt** und **tudscrbook** `declaration=chapter`.

Abhängig von der gewählten Gliederungsebene der Überschrift wird das Verhalten für das Setzen eines Eintrages ins Inhaltsverzeichnis festgelegt. Normalerweise wird nur für Überschriften in Form einer Gliederungsebene ein Eintrag der Erklärung ins Inhaltsverzeichnis erstellt, für `declaration=true` und `declaration=false` geschieht dies standardmäßig nicht. Mit folgenden Werten kann das voreingestellte Verhalten überschrieben werden.

`declaration=notoc/nottotoc`

Die Erklärung wird definitiv nicht ins Inhaltsverzeichnis eingetragen.

`declaration=toc/totoc`

Unabhängig von der Wahl der Überschrift erhält jede Erklärung einen nicht nummerierten Eintrag im Inhaltsverzeichnis auf der obersten Gliederungsebene der verwendeten Dokumentklasse (`\chapter` oder `\section`).

v2.02

`declaration=tocleveldown/leveldown/totocleveldown`

Der Inhaltsverzeichniseintrag wird eine Stufe unter der obersten Gliederungsebene (`\section` oder `\subsection`) erzeugt.

v2.04

`declaration=tocmultiple/totocmultiple/tocaggregate/totocaggregate`

Es wird ein *einzig*er Inhaltsverzeichniseintrag für *alle* Erklärungen erstellt.

v2.02

Normalerweise werden automatische Kolumnentitel abhängig von der Gliederungsebene der Überschrift gesetzt, falls diese über die KOMA-Script-Option `automark` aktiviert sind. Werden manuelle Kolumnentitel genutzt, müssen diese auch für die Erklärungen manuell gesetzt werden. Mit `declaration=markboth` lässt sich unabhängig davon das Setzen der Kolumnentitel auf linken und rechten Seiten forcieren, wobei hierfür der Titel der Überschrift genutzt wird.

`declaration=markboth`

Unabhängig von der Verwendung manueller oder automatischer Kolumnentitel werden diese auf rechten sowie linken Seiten mit den Bezeichnern `\confirmationname` beziehungsweise `\blockingname` gesetzt.

`declaration=nomarkboth`

Die Einstellung für manuelle oder automatische Kolumnentitel werden beachtet.

Für `\declaration` respektive `\confirmation` und `\blocking` sowie die `declarations`-Umgebung lässt sich mit dem Parameter `markboth` ein beliebiger Kolumnentitel setzen.

Die folgenden Einstellungen zur Positionierung der Erklärungen haben lediglich Auswirkungen, wenn die Überschrift der Erklärung *nicht* im Form eines Kapitels ausgegeben und mit der KOMA-Script-Option `titlepage=true` eine Titelseite verwendet wird.

`declaration=single/one/simple`

Jede Erklärung wird auf einer separaten Seite beziehungsweise im zweispaltigen Satz in einer neuen Spalte ausgegeben.

v2.02

`declaration=multiple/multi/all/aggregate`

Erklärungen, welche in der `declarations`-Umgebung mit den Befehlen `\confirmation`, `\blocking` und `\declaration` oder außerhalb dieser mit `\declaration` gesetzt wurden, werden direkt nacheinander auf der gleichen Seite ausgegeben, wenn ausreichend Platz auf dieser vorhanden sein sollte. Ist die Klassenoption `twocolumn` aktiviert, erfolgt die Ausgabe aller Erklärungen ohne Spaltenumbruch.

`declaration=fill/fil/vfil/vfill`

Alle Erklärungen auf einer Ausgabeseite werden vertikal zentriert. Für den zweispaltigen Satz (Klassenoption `twocolumn`) steht diese Einstellung nicht zur Verfügung.

`declaration=nofill/nofil/novfil/novfill`

Die Ausgabe erfolgt wie im normalen Fließtext auch.

v2.02  
v2.04

```
\begin{declarations}[<Sprache> || <Parameterliste>]
. . .
\end{declarations}
\nextdeclaration[<Sprache> || <Parameterliste>]{<Überschrift>}{<Erklärung>}
language=<Sprache> (Parameter)
markboth=<Einstellung> (Parameter)
pagestyle=<Seitenstil> (Parameter)
columns=<Anzahl> (Parameter)
option=<Einstellung> (Parameter)
supporter=<Unterstützer> (Parameter)
place=<Ort> (Parameter)
closing=<Ende> (Parameter)
company=<Firma> (Parameter)
```

Für Selbstständigkeitserklärung und Sperrvermerk sollte im einfachsten Fall `\declaration` beziehungsweise `\confirmation` und `\blocking` verwendet werden. Sobald diese jedoch in

anderer Reihenfolge, mehrfacher Ausführung, unterschiedlichen Sprachen oder um zusätzliche Erklärungen ergänzt werden, bietet die `declarations`-Umgebung entsprechende Freiheiten.

Innerhalb dieser Umgebung können Selbstständigkeitserklärung und Sperrvermerk mit dem Befehl `\declaration` direkt nacheinander folgend beziehungsweise mit `\confirmation` und `\blocking` auch separat ausgegeben werden. Dies kann in beliebiger Reihenfolge und auch mehrmals geschehen, um diese beispielsweise mehrsprachig zu setzen. Des Weiteren gibt es mit `\nextdeclaration` die Möglichkeit, eine Erklärung völlig frei zu verfassen. Dieser Befehl kann *ausschließlich* innerhalb der `declarations`-Umgebung genutzt werden, wobei im ersten Argument die gewünschte Überschrift und im zweiten der Inhalt respektive Text der Erklärung selbst angegeben werden muss.

v2.04

Die folgend beschriebenen Parameter können sowohl für die Umgebung `declarations` selbst als auch für die zuvor genannten Befehle als optionales Argument verwendet werden. Ähnlich wie die gleichnamigen Optionen sind auch die Umgebungen `abstract` und `declarations` sehr ähnlich zueinander. Deshalb werden die Erläuterungen relativ kurz gehalten. Ist ein Erklärung für einen Parameter etwas unverständlich, kann diese bei der Umgebung `abstract` in Abschnitt 2.3.1 auf Seite 44 nachgelesen werden.

Wurde das Paket **babel** geladen, kann die Sprache – sofern diese als Paketooption oder besser noch als Klassenoption angegeben wurde – mit dem Parameter `language=<Sprache>` für die `declarations`-Umgebung geändert werden. Dadurch werden die Bezeichner – unter anderem `\confirmationname` und `\blockingname` – sowie die Trennungsmuster innerhalb der Umgebung sprachspezifisch angepasst.

v2.02

Die Kolummentitel können mit `markboth` beeinflusst werden. Mit `markboth=true` werden für diese auf linker und rechter Seite `\confirmationname` beziehungsweise `\blockingname` verwendet. Automatische (`automark`) respektive manuelle (`manualmark`) Kolummentitel werden mit `markboth=false` genutzt. Mit `markboth=<Kolummentitel>` können diese direkt festgelegt werden. Wird `markboth` in irgendeiner Form genutzt, wird der Seitenstil `headings` automatisch gesetzt. Dieser lässt sich für die Umgebung mit dem Parameter `pagestyle` auch manuell angeben. Wurde das Paket **multicol** geladen, wird der Inhalt der Umgebung mit `columns=<Anzahl>` mehrspaltig gesetzt. Für `option` können alle gültigen Werte der Option `declaration` angegeben werden. Die Verwendung der weiteren Parameter `supporter` sowie `place` und `closing` ist in der Dokumentation des Befehls `\confirmation` zu finden, der Parameter `company` ist für `\blocking` erläutert.

v2.02

v2.05

```
\confirmation[<Unterstützer> || <Parameterliste>]
\confirmation* [<Parameterliste>]
supporter=<Unterstützer> (Parameter)
place=<Ort> (Parameter)
closing=<Ende> (Parameter)
language=<Sprache> (Parameter)
markboth=<Einstellung> (Parameter)
pagestyle=<Seitenstil> (Parameter)
columns=<Anzahl> (Parameter)
option=<Einstellung> (Parameter)
```

Mit diesem Befehl wird ein sprachspezifischer Standardtext für eine Selbstständigkeitserklärung ausgegeben, welcher in `\confirmationtext` gespeichert ist. Wie dieser angepasst beziehungsweise geändert werden kann, ist unter Unterkapitel 2.4 zu finden. Er kann sowohl innerhalb der `declarations`-Umgebung als auch außerhalb dieser direkt im Dokument verwendet werden.



v2.05

Wird `\confirmationtext` nicht geändert, kann dieser über das optionale Argument von `\confirmation` und die deklarierten Parameter angepasst werden. Im Standardtext der Selbstständigkeitserklärung werden sowohl der Titel als auch der Typ der Abschlussarbeit – falls dieser mit `\thesis`, `\subject{<Tabelle 2.1>}` beziehungsweise mit der Option `subjectthesis` angegeben wurde – aufgeführt. Über den Parameter `supporter` oder *zuvor* mit dem Befehl `\supporter` können weitere an der Arbeit beteiligte Personen angegeben werden. Mehrere zu nennende Personen sind auch hier durch `\and` zu trennen. Das Feld der Unterstützer kann auch mit dem bloßen optionalen Argument ohne die Angabe eines Parameters angepasst werden. Mit der Sternversion `\confirmation*` werden als Unterstützer die mit `\supervisor{<Name(n)>}` definierten Betreuer der Arbeit angegeben.

Nach dem eigentlichen Text der Selbstständigkeitserklärung wird der mit `place` beziehungsweise `\place` angegebene Ort sowie das mit `\date` eingestellte Datum ausgegeben. Als Voreinstellung ist für den Ort „Dresden“ gewählt. Danach folgen – mit etwas vertikalem Leer-  
raum für die notwendige Unterschrift – der Autor oder die Autoren, angegeben durch den Befehl `\author`. Soll anstelle dessen etwas anderes nach dem Text der Selbstständigkeitserklärung gesetzt werden, kann dies mit dem Parameter `closing` oder *zuvor* mit dem Befehl `\confirmationclosing` angepasst werden. Die Parameter `language`, `markboth`, `pagestyle`, `columns` und `option` entsprechen in ihrem Verhalten denen der `declarations`-Umgebung.

v2.02

```
\blocking[<Firma> || <Parameterliste>]  
company=<Firma> (Parameter)  
language=<Sprache> (Parameter)  
markboth=<Einstellung> (Parameter)  
pagestyle=<Seitenstil> (Parameter)  
columns=<Anzahl> (Parameter)  
option=<Einstellung> (Parameter)
```

Beim Sperrvermerk verhält es sich äquivalent zur Selbstständigkeitserklärung. Es wird der in `\blockingtext` hinterlegte Standardtext in der gewählten Sprache ausgegeben. Dieser kann – wie in Unterkapitel 2.4 beschrieben – geändert werden. Der Befehl `\blocking` kann sowohl innerhalb der Umgebung `declarations` als auch direkt im Dokument verwendet werden.

In seiner ursprünglichen Definition, kann er im optionalen Argument über die deklarierten Parameter angepasst werden. Im Standardtext des Sperrvermerks werden sowohl der Titel als auch der Typ der Abschlussarbeit – falls dieser mit `\thesis`, `\subject{<Tabelle 2.1>}` respektive mit der Option `subjectthesis` angegeben wurde – aufgeführt. Mit `company` oder *vorher* mit `\company` kann zusätzlich eine im Sperrvermerk zu nennende Firma oder ähnliches angegeben werden. Dieses Feld lässt sich auch direkt im optionalen Argument ohne die Verwendung eines Parameters definieren. Die weiteren Parameter `language`, `markboth`, `pagestyle`, `columns` und `option` entsprechen in ihrem Verhalten denen der `declarations`-Umgebung.

v2.02  
v2.05

```
\declaration[<Parameterliste>]  
\declaration*[<Parameterliste>  
language=<Sprache> (Parameter)  
markboth=<Einstellung> (Parameter)  
pagestyle=<Seitenstil> (Parameter)  
columns=<Anzahl> (Parameter)  
option=<Einstellung> (Parameter)  
supporter=<Unterstützer> (Parameter)  
place=<Ort> (Parameter)  
closing=<Ende> (Parameter)  
company=<Firma> (Parameter)
```

Dieser Befehl gibt die Selbstständigkeitserklärung und den Sperrvermerk direkt aufeinanderfolgend aus. Dabei werden die Einstellungen zur Positionierung der einzelnen Erklärungen, welche über die Zuweisungen `declaration=single` beziehungsweise `declaration=multiple` sowie `declaration=fill` respektive `declaration=nofill` erfolgen, beachtet. Er kann sowohl innerhalb der `declarations`-Umgebung als auch außerhalb dieser direkt im Dokument verwendet werden und akzeptiert im optionalen Argument dabei alle für die `declarations`-Umgebung beschriebenen Parameter. Die Sternversion erzwingt für die Selbstständigkeitserklärung eine Angabe der mit `\supervisor{<Name(n)>}` definierten Betreuer in dieser.

v2.05

```
\supporter{<Unterstützer>  
\place{<Ort>  
\confirmationclosing{<Ende>  
\company{<Firma>}
```

Diese Makros ändern – im Gegensatz zu den Parametern von `\confirmation` und `\blocking` – die entsprechenden Feldwerte `global`. Damit lässt sich die *mehrfache* Angabe eines Parameters vermeiden, wenn beispielsweise eine Erklärung in unterschiedlichen Sprachen erzeugt wird.

### 2.3.3 Lesezeichen

```
tudbookmarks=<Ein-Aus-Wert>  
\tudbookmark[<Ebene>]{<Text>}{<Ankername>}
```

Voreinstellung: `true`

Diese Option wird wirksam, wenn `hyperref` geladen wurde. Es werden für die Umschlag- und Titelseite, das Inhaltsverzeichnis sowie – bei der Verwendung des Paketes `tudscrsupervisor` – die Aufgabenstellung Lesezeichen oder auch Outline-Einträge im PDF-Dokument erzeugt.

`tudbookmarks=false/off/no`

Es erfolgt kein Eintrag von ergänzenden Lesezeichen.

`tudbookmarks=true/on/yes` (Säumniswert)

Es werden automatisch zusätzliche Lesezeichen eingetragen.

Der Befehl `\tudbookmark` arbeitet prinzipiell in der gleichen Weise wie `\pdfbookmark` aus `hyperref`. Die Lesezeichen werden jedoch nur bei aktivierte Option `tudbookmarks` generiert.

## 2.4 Sprachabhängige Bezeichner

Durch KOMA-Script werden Befehle, mit denen sich sprachabhängige Bezeichner erzeugen oder ändern lassen, zur Verfügung gestellt. Diese werden durch das TUD-Script-Bundle ge-

nutzt, um lokalisierte Begriffe für die Sprachen *Englisch* und *Deutsch* bereitzustellen. Ein Großteil davon betrifft Bezeichnungen für Felder auf der Titelseite (Abschnitt 2.2.4). Hierfür wird `\providecaptionname{<Sprache>}{<Makro>}{<Inhalt>}` verwendet, wobei `<Sprache>` dem geladenen Sprachpaket – normalerweise das Paket **babel** – bekannt sein muss.

Sollte der Anwender die im Folgenden erläuterten oder auch andere Bezeichner, welche von einem beliebigen (Sprach-)Paket bereitgestellt werden, ändern wollen, ist hierfür der Befehl `\renewcaptionname{<Sprache>}{<Makro>}{<Inhalt>}` zu verwenden. Es sollte natürlich dabei eine `<Sprache>` angegeben werden, welche im Dokument durch **babel** oder ein anderes Sprachpaket verwendet wird, beispielsweise `ngerman` oder `english`.

Die Makros der Bezeichner und deren Verwendung werden folgend kurz beschrieben und tabellarisch aufgeführt. Dabei wurde versucht, alle Befehle der Bezeichner für bestimmte Begriffe auf `\ . . .name` und beschreibende Texte auf `\ . . .text` enden zu lassen.

v2.02  
v2.05

```
\refereename (Bezeichner)
\refereeothername (Bezeichner)
\advisorname (Bezeichner)
\advisorothername (Bezeichner)
\supervisorname (Bezeichner)
\supervisorothername (Bezeichner)
\professorname (Bezeichner)
\professorothername (Bezeichner)
```

Diese sprachabhängigen Begriffe sind die Bezeichner für die Titelseitenfelder von Betreuer (`\supervisor`), Gutachter (`\referee`) und Fachreferent (`\advisor`). Soll innerhalb eines dieser Felder mehr als eine Person angegeben werden, so sind die Einzelpersonen jeweils mit dem Befehl `\and` voneinander zu trennen. In diesem Fall werden alle nach der erstgenannten folgenden Personen durch den Bezeichner `\ . . .othername` ergänzt.

v2.02  
v2.05

Bei den Bezeichnung wird unterschieden, ob eine oder mehrere Personen angegeben wurden. Wird lediglich eine Person genannt, so ist eine Unterscheidung nicht notwendig und es wird der Singular genutzt. Werden jedoch zwei oder mehr Personen angegeben, so wird geprüft, ob der dazugehörige Bezeichner für die Zweitperson (`\ . . .othername`) definiert ist. Falls dies so ist, wird die alternative Bezeichnung für die erstgenannte Person verwendet, andernfalls wird der Plural des Bezeichners verwendet. Dies betrifft alle Felder, die über `\referee`, `\advisor`, `\supervisor` oder `\professor` angegeben wurden.

Bezeichner	Deutsch	Englisch
<code>\supervisorname</code>	(Erst-)Betreuer	(First) Supervisors(s)
<code>\supervisorothername</code>	<leer>	<empty>
<code>\refereename</code>	(Erst-)Gutachter	(First) Referee(s)
<code>\refereeothername</code>	Zweitgutachter	Second referee
<code>\advisorname</code>	(Erster) Fachreferent(en)	(First) Advisor(s)
<code>\advisorothername</code>	<leer>	<empty>
<code>\professorname</code>	Betreuende(r) Hochschullehrer	Supervising professor(s)
<code>\professorothername</code>	<leer>	<empty>

```

\dissertationname (Bezeichner)
\diplomathesisname (Bezeichner)
\masterthesisname (Bezeichner)
\bachelorthesisname (Bezeichner)
\studentthesisname (Bezeichner)
\studentresearchname (Bezeichner)
\projectpapername (Bezeichner)
\seminarpapername (Bezeichner)
\termpapername (Bezeichner)
\researchname (Bezeichner)
\logname (Bezeichner)
\internshipname (Bezeichner)
\reportname (Bezeichner)

```

Diese Bezeichner dienen zur Typisierung speziell für eine Abschlussarbeit. Wie diese genutzt werden können, ist bei der Erläuterung von `subjectthesis` beziehungsweise `\thesis` und `\subject` in Abschnitt 2.2.4 auf Seite 37 zu finden.

Bezeichner	Deutsch	Englisch
<code>\dissertationname</code>	Dissertation	Dissertation
<code>\diplomathesisname</code>	Diplomarbeit	Diploma Thesis
<code>\masterthesisname</code>	Master-Arbeit	Master Thesis
<code>\bachelorthesisname</code>	Bachelor-Arbeit	Bachelor Thesis
<code>\studentresearchname</code>	Großer Beleg	Student Research Project
<code>\projectpapername</code>	Projektarbeit	Project Paper
<code>\seminarpapername</code>	Seminararbeit	Seminar Paper
<code>\researchname</code>	Forschungsbericht	Research Report
<code>\logname</code>	Protokoll	Log
<code>\internshipname</code>	Praktikumsbericht	Internship Report
<code>\reportname</code>	Bericht	Report

v2.02

```
\graduationtext (Bezeichner)
```

Wurde erkannt, dass das aktuelle Dokument eine Abschlussarbeit ist, so kann der zu erlangende akademische Grad mit dem Befehl `\graduation` angegeben werden. Bei dessen Ausgabe auf dem Titel wird dabei der entsprechende Text dazu angegeben.

Bezeichner	Deutsch	Englisch
<code>\graduationtext</code>	zur Erlangung des akademischen Grades	to achieve the academic degree

```

\datetext (Bezeichner)
\defensedatetext (Bezeichner)

```

Wird mit `\date` das (Abgabe-)Datum und mit `\defensedate` ein Datum der Verteidigung für eine Abschlussarbeit angegeben, so werden auch diese Felder durch einen Text beschrieben.

Bezeichner	Deutsch	Englisch
<code>\datetext</code>	Eingereicht am	Submitted on
<code>\defensedatetext</code>	Verteidigt am	Defended on

v2.02

`\dateofbirthtext` (Bezeichner)  
`\placeofbirthtext` (Bezeichner)  
`\matriculationnumbername` (Bezeichner)  
`\matriculationyearname` (Bezeichner)  
`\coursename` (Bezeichner)  
`\disciplinename` (Bezeichner)

Werden für den Autor oder die Autoren mit dem entsprechenden Befehl das Geburtsdatum (`\dateofbirth`), der Geburtsort (`\placeofbirth`), der Studiengang (`\course`), die Studienrichtung (`\discipline`) oder auch die Matrikelnummer (`\matriculationnumber`) und/oder das Immatrikulationsjahr (`\matriculationyear`) angegeben, werden sowohl auf der Titelseite als auch auf der gegebenenfalls mit **tudscr**supervisor erstellten Aufgabenstellung die dazugehörigen Bezeichner vorangestellt. Auf dem Titel werden diese dabei mit dem durch `\titledelimiter` gegebenen Trennzeichen vom eigentlichen Feld abgegrenzt.

Bezeichner	Deutsch	Englisch
<code>\dateofbirthtext</code>	Geboren am	Born on
<code>\placeofbirthtext</code>	in	in
<code>\matriculationnumbername</code>	Matrikelnummer	Matriculation number
<code>\matriculationyearname</code>	Immatrikulationsjahr	Matriculation year
<code>\coursename</code>	Studiengang	Course
<code>\disciplinename</code>	Studienrichtung	Discipline

`\coverpagename` (Bezeichner)  
`\titlepagename` (Bezeichner)

Diese beiden Bezeichner werden bei aktivierter `tudbookmarks` für das Eintragen von Lesezeichen in ein PDF-Dokument genutzt.

Bezeichner	Deutsch	Englisch
<code>\coverpagename</code>	Umschlagseite	Cover page
<code>\titlepagename</code>	Titelblatt	Title page

`\abstractname` (Bezeichner)

Dieser Bezeichner wird für die Klasse **tudscrbook** definiert, da selbiger von KOMA-Script für die Buchklasse nicht vorgesehen wird.

Bezeichner	Deutsch	Englisch
<code>\abstractname</code>	Zusammenfassung	Abstract

v2.02

`\confirmationname` (Bezeichner)  
`\blockingname` (Bezeichner)

Es werden die Bezeichnungen für Selbstständigkeitserklärung und Sperrvermerk für die dazugehörigen Überschriften definiert.

Bezeichner	Deutsch	Englisch
<code>\confirmationname</code>	Selbstständigkeitserklärung	Statement of authorship
<code>\blockingname</code>	Sperrvermerk	Restriction note

`\confirmationtext` (Bezeichner)`\blockingtext` (Bezeichner)

Die Texte der Erklärungen selbst sind derart aufgebaut, dass sie in Abhängigkeit von den angegebenen Informationen unterschiedlich ausgeführt werden. Innerhalb der Selbstständigkeitserklärung (`\confirmation`) werden gegebenenfalls die Felder für den Titel (`\title`) und die Typisierung der Abschlussarbeit sowie die angegebenen Unterstützer<sup>34</sup> beachtet. Für den Sperrvermerk (`\blocking`) wird neben dem Titel (`\title`) optional außerdem noch das Feld der externen Firma<sup>35</sup> verwendet. Der Vollständigkeit halber werden im Folgenden noch die Texte für die Selbstständigkeitserklärung und den Sperrvermerk aufgeführt – allerdings lediglich die deutschsprachige Version. Dabei werden alle möglichen Felder angezeigt.

#### Bezeichner `\confirmationtext`

Hiermit versichere ich, dass ich das vorliegende Dokument mit dem Titel `<Title>` selbstständig und ohne unzulässige Hilfe Dritter verfasst habe. Es wurden keine anderen als die in diesem Dokument angegebenen Hilfsmittel und Quellen benutzt. Die wörtlichen und sinngemäß übernommenen Zitate habe ich als solche kenntlich gemacht. Während der Anfertigung dieses Dokumentes wurde ich nur von folgenden Personen unterstützt:

`<Vorname Nachname>`

`<Vorname Nachname>`

Zusätzliche Personen waren an der geistigen Herstellung des vorliegenden Dokumentes nicht beteiligt. Mir ist bekannt, dass die Nichteinhaltung dieser Erklärung zum nachträglichen Entzug des Hochschulabschlusses führen kann.

#### Bezeichner `\blockingtext`

Dieses Dokument mit dem Titel `<Title>` enthält vertrauliche Informationen, offengelegt durch `<Firma>`. Veröffentlichungen, Vervielfältigungen und Einsichtnahme – auch nur auszugsweise – sind ohne ausdrückliche Genehmigung durch `<Firma>` nicht gestattet, ebenso wie Veröffentlichungen über den Inhalt dieses Dokumentes. Es ist nur dem Betreuer an der Technischen Universität Dresden, den Gutachtern sowie den Mitgliedern des Prüfungsausschusses zugänglich zu machen.

Soll eine der Erklärungen geändert und dabei der Inhalt eines Feldes genutzt werden, lässt sich hierfür `\getfield` verwenden.<sup>36</sup> Gegebenenfalls ist die Definition von `\and` anzupassen.

`\listingname` (Bezeichner)`\listlistingname` (Bezeichner)

Sollte ein Paket zur Einbindung von externem Quelltext – beispielsweise das Paket `listings` – verwendet werden, so werden diese Bezeichnungen für Quelltextausschnitte und das Quelltextverzeichnis verwendet.

Bezeichner	Deutsch	Englisch
<code>\listingname</code>	Quelltext	Listing
<code>\listlistingname</code>	Quelltextverzeichnis	List of Listings

<sup>34</sup>`\confirmation[supporter=<Unterstützer>]` oder `\supporter{<Unterstützer>}`

<sup>35</sup>`\blocking[company=<Firma>]` oder `\company{<Firma>}`

<sup>36</sup>Titel: `\getfield{title}` → `<Title>`, Art der Abschlussarbeit: `\getfield{thesis}` → `<Abschlussarbeit>`, Autor: `\getfield{author}` → `<Vorname Nachname>`, Firma: `\getfield{company}` → `<Firma>` sowie ebenfalls Unterstützer: `\getfield{supporter}` → `<Vorname Nachname>`

## 2.5 Kompatibilitätseinstellungen zu früheren Versionen

Bei der Entwicklung von TUD-Script lässt es sich nicht immer vermeiden, dass Verbesserungen sowie Korrekturen an den Klassen und Paketen zu Änderungen am Ergebnis der Ausgabe führen, insbesondere bei Umbruch und Layout. Für bereits archivierte Dokumente, welche mit einer früheren Version erstellt wurden ist dies jedoch bei einer erneuten Kompilierung unter Umständen eher unerwünscht.

v2.03 `tudscrver=<Version>` Voreinstellung: `last`

Mit dieser Option wird es möglich, auf das (Umbruch-)Verhalten einer älteren respektive früheren Version von TUD-Script umzuschalten, um nach der Kompilierung das erwartete Ergebnis zu erhalten. Neue Möglichkeiten, die sich nicht auf den Umbruch oder das Layout auswirken, sind auch für den Fall verfügbar, dass per Option die Kompatibilität zu einer älteren Version ausgewählt wurde.

Bei der Angabe einer unbekanntenen Version als Wert wird eine Warnung ausgegeben und `tudscrver=first` angenommen. Mit `tudscrver=last` wird die jeweils aktuell verfügbare Version ausgewählt und folglich auf die zukünftige Kompatibilität des Dokumentes zu der aktuell genutzten Version verzichtet. Dieses Verhalten entspricht der Voreinstellung. Es ist zu beachten, dass die Nutzung von `tudscrver` nur als Klassenoption möglich ist.

v2.03 `tudscrver=first/2.02`

Der Satzspiegel im Layout des Corporate Designs (*siehe cedgeometry*) wurde in der Version v2.03 leicht geändert. Der obere Seitenrand wurde verkleinert, der untere im gleichen Maße vergrößert. Der verfügbare Textbereich ist folglich identisch. Bei der Aktivierung des DRESDEN-concept-Logos im Fußbereich der Seite (*siehe ddcfoot*) wird im Gegensatz zur Version v2.02 der gleiche Satzspiegel genutzt. Mit `tudscrver=first` kann dieses Verhalten deaktiviert werden.

v2.04 `tudscrver=2.03`

Seit der Version v2.04 werden mehrere Längen in Abhängigkeit der gewählten Schriftgröße definiert. Mit der Wahl `tudscrver=2.03` wird diese Funktionalität deaktiviert, wobei hierfür lediglich die TUD-Script-Option `relspacing=false` aktiviert wird.

v2.05 `tudscrver=2.04`

Mit der Version v2.05 werden die vorgegebenen Einstellungen zum Satzspiegel anhand der B-ISO/DIN-Reihe vorgenommen. Damit sind für alle Papierformate einer spezifischen ISO/DIN-Klasse die Seitenränder identisch. Mit der Wahl `tudscrver=2.04` ist der Satzspiegel von der A-ISO/DIN-Reihe abhängig, sodass die B- und C-Papierformate der gleichen Klasse größere Seitenränder erhalten, als die D- und A-Formate.

v2.06 `tudscrver=2.05`

Mit dem Wechsel der Hausschrift zu **Open Sans** ergaben sich einige Änderungen bezüglich der Erscheinung des Corporate Designs. Mit dieser Kompatibilitätseinstellung werden die alten Schriftfamilien **Univers** und **DIN Bold** (`cdo1dfont=true`, `headings=light`, `ttfont=lmodern`) aktiviert. Weiterhin wirkt sich für diese Schriftfamilien im Mathematikmodus die Option `slantedgreek=true` lediglich auf die griechischen Majuskeln aus.

`tudscrver=2.06`

Dies ist Kompatibilitätseinstellung für TUD-Script v2.06o und wird für zukünftige Änderungen bereits vorgehalten. Soll ein mit der momentan aktuellen Version erzeugtes Dokument auch mit einer späteren Version von TUD-Script nach einem  $\LaTeX$ -Lauf das gleiche Ausgabeergebnis liefern, muss dies mit `tudscrver=2.06` angegeben werden.

`tudscrver=last`

Es werden keine Kompatibilitätseinstellungen für das Dokument vorgenommen. Mit einer späteren Version von TUD-Script kann ein anderes Umbruchverhalten innerhalb des Dokumentes auftreten. Dies ist die Standardeinstellung.



## Kapitel 3

# Die Posterklasse

Ergänzend zu den Hauptklassen, welche für das Setzen von Dokumenten im Corporate Design der Technischen Universität Dresden angeboten werden, wird die Klasse **tudscrposter** bereitgestellt. Mit dieser wird das Erstellen von Postern im gleichen Layout mit  $\LaTeX$  ermöglicht. Die Basis hierfür ist **tudscrartcl** und *fast* alle durch diese Klasse angebotenen Befehle und Optionen können gleichermaßen mit **tudscrposter** verwendet werden. Ein Minimalbeispiel zur Verwendung der Klasse ist in Unterkapitel 6.7 auf Seite 95 zu finden.

Der größte Unterschied zu den Hauptklassen ist insbesondere ein vereinfachter Titel. Eine Umschlagseite steht für **tudscrposter** nicht zur Verfügung, die entsprechende Option `cdcover` sowie der dazugehörige Befehl `\makecover` sind nicht definiert. Der Titel selbst kann mit `\maketitle` lediglich als Titelpf gesetzt werden, eine separate Titelseite existiert nicht. Aus diesem Grund sind mit `\title`, `\subtitle`, `\subject` und `\titlehead` auch nur eine reduzierte Anzahl an Befehlen für den Titel verfügbar, die wie gewohnt genutzt werden können. Prinzipiell lassen sich auch noch die beiden Befehle für den Schmutztitel `\extratitle` sowie `\frontispiece` nutzen, wobei deren Verwendung für ein Poster eher fraglich ist.

Alle weiteren in Abschnitt 2.2.4 vorgestellten Befehle und Optionen sind für **tudscrposter** nicht definiert. Dies betrifft zum einen sowohl das Schriftelement `thesis` als auch die Befehle `\titledelimiter`, `\thesis`, `\referee`, `\advisor`, `\graduation` und `\defensedate`. Zum anderen stehen die Makros für ergänzende Autorenangaben `\dateofbirth`, `\placeofbirth`, `\matriculationyear` und `\matriculationnumber` wie auch die Option `subjectthesis` *nicht* zur Verfügung. Die durch KOMA-Script für eine Titelseite bereitgestellten Befehle `\date`, `\publishers` und `\dedication` sowie `\uppertitleback` und `\lowertitleback` haben bei der Klasse **tudscrposter** keinerlei Funktionalität. Die Befehle `\author` und `\authormore` existieren weiterhin, werden allerdings nicht für den Titel wohl jedoch für den speziellen Fußbereich eines Posters verwendet, welcher in Unterkapitel 3.2 weiterführend beschrieben wird.

Neben der signifikanten Vereinfachung des Titels entfallen für die Klasse **tudscrposter** einige weitere Befehle und Umgebungen. Namentlich sind dies die Umgebung `tudpage`, die Optionen `headingsvskip` und `pageheadingsvskip` sowie alle zu Selbstständigkeitserklärung und Sperrvermerk gehörigen Elemente, wie die Option `(declaration)`, die Umgebung `(declarations)` und die Befehle `(\declaration, \confirmation, \blocking)`. Die Umgebung `abstract` kann weiterhin genutzt werden, allerdings kann mit der Option `abstract` lediglich noch die Gliederungsebene der Überschrift angepasst werden.

### 3.1 Layout und Formatierung eines Posters

Die augenscheinlichsten Einstellungen für die Gestaltung eines Posters sind sicherlich das verwendete Papierformat und die farbliche Ausprägung sowie die Auswahl der Schriftart und deren Größe. Als Grundeinstellung für die Klasse **tudscrposter** sind die Schriften des Corporate Designs der Technischen Universität Dresden aktiviert. Diese lassen sich – wie auch bei den Hauptklassen – anpassen. Weitere Informationen hierzu sind der Erläuterung zur Option `cdfont` in Abschnitt 2.1.1 auf Seite 16 zu entnehmen.

Nachfolgend wird kurz erläutert, wie sich allgemeine Formatierung eines mit **tudscrposter** erstellten Posters anpassen lässt. Der Inhalt kann völlig frei gestaltet werden, es gibt hierfür bisher keinerlei von TUD-Script vordefinierte Befehle und Optionen, welche die Inhalte in ein bestimmtes Layout übersetzen.

**Tabelle 3.1** Empfohlene Kombinationen für die Wahl von Papierformat (paper) und Schriftgröße (fontsize)

a) Einspaltiges Layout

		Klasse						
		6	5	4	3	2	1	0
Reihe	min...max	min...max	min...max	min...max	min...max	min...max	min...max	min...max
D	05...07pt	06...09pt	10...14pt	14...20pt	20...29pt	28...40pt	40...60pt	
A	06...08pt	07...10pt	11...16pt	16...23pt	23...33pt	32...46pt	45...66pt	
C	07...09pt	08...11pt	12...18pt	18...26pt	26...37pt	36...52pt	50...72pt	
B	08...10pt	09...12pt	13...20pt	20...29pt	29...41pt	40...58pt	55...78pt	

b) Zweispaltiges Layout

		Klasse			
		3	2	1	0
Reihe	min...max	min...max	min...max	min...max	min...max
D	07...10pt	10...14pt	14...19pt	19...28pt	
A	08...11pt	11...16pt	16...22pt	22...32pt	
C	09...12pt	12...18pt	18...25pt	25...36pt	
B	10...13pt	13...20pt	20...28pt	28...40pt	

c) Dreispaltiges Layout

		Klasse	
		1	0
Reihe	min...max	min...max	min...max
D	07...10pt	10...14pt	
A	08...11pt	11...16pt	
C	09...12pt	12...18pt	
B	10...13pt	13...20pt	

### 3.1.1 Die Wahl von Papierformat und Schriftgröße

Achtung!

Die Festlegung von *Papierformat* und *Schriftgröße* ist essentiell für das Erstellen eines Posters und sollten *immer* vorgenommen werden. *Beide Einstellungen müssen zwingend als Klassenoption erfolgen.* Bei der Schriftgrößenauswahl ist darauf zu achten, ob der Satz des Posters ein- oder mehrspaltig erfolgen soll. Für letzteres Unterfangen ist die `multicols`-Umgebung aus dem Paket `multicol` sehr empfehlenswert.

Zur Festlegung des Papierformats ist die KOMA-Script-Option `paper=<Einstellung>` zu verwenden. Dabei lassen sich mit `paper=<Format>` unter anderem die gängigen Klassen der ISO/DIN-Reihen A bis D als auch Quer- oder Längsformat auswählen. Ein beliebiges Format kann mit der Einstellung `paper=<Höhe> : <Breite>` gewählt werden. Für zusätzliche Hinweise ist das KOMA-Script-Handbuch<sup>37</sup> zu Rate zu ziehen.

Passend zum ausgewählten Papierformat sowie der gewünschten Anzahl an Textspalten des Posters sollte unbedingt die Schriftgröße mit `fontsize=<Schriftgröße>` angegeben werden. Für eine passend abgestimmte Auswahl von Papierformat und Schriftgröße ist Tabelle 3.1 als Referenz zu nutzen. Sollten Sie aufgrund der Schriftgrößenänderung eine oder mehrere Warnungen vom Typ

```
Font shape 'T1/cmr/m/n' in size <...> not available
```

erhalten, so beachten Sie bitte die Hinweise aus Unterkapitel 9.24.

<sup>37</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>

### 3.1.2 Die Gestalt eines Posters

Die Festlegung der Farbausprägung eines Posters erfolgt mit der Option `cd`, welche nachfolgend beschrieben wird. Dabei kann aus einigen Varianten zur Farbgestaltung gewählt werden. Sollte keiner dieser vordefinierten Werte das gewünschte Layout zur Verfügung stellen, lässt sich dieses mit den Optionen `cdhead` und `cdfoot` sowie `cdtitle`, `cdpart` und `cdsection` nachträglich noch genauer anpassen.

`cd=<Einstellung>` Voreinstellung: `bicolor`

Äquivalent zu den TUD-Script-Hauptklassen wird mit dieser Option die Verwendung des Corporate Designs der Technischen Universität Dresden für das Poster festgelegt. Sie hat Einfluss auf die Farbgestaltung der Gliederungsüberschriften sowie des Seitenstils, welcher standardmäßig auf `empty.tudheadings` gesetzt wird.

`cd=false/off/no`

Hiermit wird das Corporate Design komplett deaktiviert und es werden keine spezifischen Einstellungen für ein Poster vorgenommen. Lediglich der Seitenstil wird auf `empty` festgelegt.

`cd=true/on/yes/nocolor/monochrome` (Säumniswert)

Es wird schwarze Schrift für Überschriften und den Seitenkopf verwendet. Der Fußbereich wird nicht farbig akzentuiert.

`cd=lightcolor/pale`

Die Einstellung entspricht weitestgehend der Option `cd=true`, allerdings wird die primäre Hausfarbe HKS41 für Kopf sowie Fuß und die Überschriften genutzt.

`cd=barcolor`

Zusätzlich zur vorherigen Einstellung wird außerdem der Querbalken farbig abgesetzt.

`cd=bicolor/color/fullcolor`

Der Kopf wird mit einem farbigen Hintergrund in der primären Hausfarbe HKS41 gesetzt, der Querbalken wird farbig abgesetzt. Ebenso wird für alle Überschriften die Hausfarbe verwendet, der Fußbereich erhält ebenfalls einen farbigen Hintergrund.

`backcolor=<Einstellung>` Voreinstellung: `true`

Mit dieser Option kann die Hintergrundfarbe eines Posters definiert werden.

`backcolor=false/off/no/nocolor`

Es wird keine Farbe festgelegt, der Hintergrund erscheint weiß.

`backcolor=true/on/yes/color` (Säumniswert)

Der Seitenhintergrund wird in der primären Hausfarbe HKS41 gewählt.

`backcolor=<Farbe>`

Die angegebene `<Farbe>` wird als Hintergrund für das Poster genutzt.

`bleedmargin=<Längenwert>` Voreinstellung: `0.2in`

Soll das Poster in einem Papierformat gedruckt werden, welches anschließend noch auf das Zielformat zugeschnitten wird, weil beispielsweise ein randloses Drucken nicht möglich ist, kann diese Option genutzt werden, um die farbigen Elemente des Layouts in den Bereich der Beschnittzugabe respektive Überfüllung zu vergrößern. Damit ist ein „Zuschneiden in die Farbe“ sehr einfach und ohne große Probleme realisierbar.

Die von der Einstellung `bleedmargin=<Längenwert>` abhängigen Elemente sind zum einen Kopf- und Fußbereich, beeinflusst durch die Optionen `cdhead` und `cdfoot`. Werden diese farbig gesetzt, so werden diese um den angegebenen `<Längenwert>` über das gewünschte

Zielformat hinaus vergrößert. Zum anderen wird auch der mit `backcolor` gegebenenfalls eingestellte, farbige Seitenhintergrund erweitert. Wie sich der Entwurf eines Posters in einem bestimmten Zielformat auf einem übergroßem Papierbogen tatsächlich realisieren lässt, wird in Unterkapitel 9.21 auf Seite 124 exemplarisch dargestellt.

## 3.2 Felder für den Fußbereich

Der Fußbereich eines Posters kann mit `\footcontent` eigens und frei definiert werden. Geschieht dies nicht, wird standardmäßig ein vordefinierte Fuß gesetzt, welcher Angaben von bestimmten Feldern ausgibt, die insbesondere als Kontaktinformationen gedacht sind. Welche das im Einzelnen sind, wird nachfolgend erläutert. Die farbliche Ausprägung des Fußes wird durch die Option `cdfoot` festgelegt.

```
\faculty[<Fußzeile>]{<Fakultät>}
\department[<Fußzeile>]{<Einrichtung>}
\institute[<Fußzeile>]{<Institut>}
\chair[<Fußzeile>]{<Lehrstuhl>}
```

Die mit diesen Befehlen gemachten Angaben werden nicht nur im Kopf sondern zusätzlich auch im linken Teil des Fußbereichs ausgegeben. Sollen diese für den Fußbereich angepasst werden, lässt das optionale Argument hierfür verwenden, wobei die Angabe eines leeren optionalen Argumentes das Feld für den Fuß komplett unterdrückt. Vor allen Angaben wird der Bezeichner `\contactname` in fetter Schrift ausgegeben.

```
\professor{<Name>}
```

Zusätzlich zu den Angaben der Einrichtung kann mit `\professor` der aktuelle Inhaber der genannten Professur im linken Fußbereich angegeben werden.

```
\author{<Autor(en)>}
\contactperson{<Name(n)>}
\authormore{<Autorenzusatz>}
\course{<Studiengang>}
\discipline{<Studienrichtung>}
\office{<Adresse/Gebäude>}
\telephone{<Telefonnummer>}
\telefax{<Telefaxnummer>}
```

Der oder die mit `\author` angegebenen Autoren werden im rechten Teil des Fußbereichs (nacheinander) ausgegeben, mehrere Autoren sind mit `\and` voneinander zu trennen. Die Befehle `\authormore`, `\course` und `\discipline` sowie `\office`, `\telephone`, `\telefax` und `\emailaddress` können für zusätzliche Angaben zu jedem Autor innerhalb des Argumentes von `\author` verwendet werden. Vor der Ausgabe aller Autoreninformationen wird der Bezeichner `\authorname` in fetter Schrift gesetzt. Ohne die Angabe von `\author` erfolgt keine Ausgabe.

Danach folgen alle mit `\contactperson` gemachten Angaben. Auch hier ist `\and` für eine Trennung mehrerer Personen zu nutzen, wobei hier lediglich die Befehle `\emailaddress`, `\office`, `\telephone` und `\telefax` nicht jedoch `\authormore` sowie `\course` und `\discipline` für zusätzliche Angaben zu verwenden sind. Bevor die Ansprechpartner ausgegeben werden, wird der Bezeichner `\contactpersonname` in fetter Schrift gesetzt. Es ist natürlich auch möglich nur Autor(en) oder Ansprechpartner anzugeben.

```
\webpage[<Einstellungen>]{<URL>}  
\webpage*{<URL>}
```

Ganz zum Schluss kann für die rechte Spalte des Fußbereichs eine Homepage angegeben werden. Wurde das Paket **hyperref** geladen, wird diese in einen Hyperlink gewandelt. Über das optionale Argument können beliebige Einstellungen an `\hypersetup` aus besagtem Paket übergeben werden. Soll die Formatierung des Eintrags manuell erfolgen, so kann die Sternversion `\webpage*` verwendet werden, wobei alle gewünschten Einstellungen innerhalb des Argumentes – gegebenenfalls in einer Gruppe – vorgenommen werden müssen.

### 3.3 Sprachabhängige Bezeichner für den Fußbereich

```
\contactname (Bezeichner)  
\authorname (Bezeichner)  
\contactpersonname (Bezeichner)
```

Wie bereits zuvor erläutert, werden diese Bezeichner in der linken respektive rechten Spalte im Fuß vor der Ausgabe der eigentlichen Felder gesetzt.

Bezeichner	Deutsch	Englisch
<code>\contactname</code>	Kontakt	Contact
<code>\authorname</code>	Autor	Author
<code>\contactpersonname</code>	Ansprechpartner	Counterpart

## Kapitel 4

# Betreuung studentischer Arbeiten

Dieses Paket stellt für das Erstellen von Aufgabenstellungen und Gutachten wissenschaftlicher Arbeiten sowie offiziellen Aushängen im Corporate Design passende Umgebungen und Befehle für den Anwender bereit. Deshalb richtet es sich vornehmlich an Mitarbeiter an der Technischen Universität Dresden, kann jedoch natürlich auch von Studenten genutzt werden.

## 4.1 Aufgabenstellung für eine wissenschaftliche Arbeit

v2.05

```
\begin{task}[<Überschrift> || <Parameterliste>]
. . .
\end{task}
headline=<Überschrift> (Parameter)
style=<Stil> (Parameter)
```

(tudpage in Abschnitt 2.2.3)

Mit der `task`-Umgebung kann eine Aufgabenstellung für eine wissenschaftliche Arbeit ausgegeben werden. Diese basiert auf der Umgebung `tudpage` und akzeptiert deshalb im optionalen Argument alle Parameter, welche bei der Beschreibung von `tudpage` in Abschnitt 2.2.3 auf Seite 32 erläutert wurden.

Für die Aufgabenstellung wird normalerweise eine Überschrift gesetzt, welche sich aus `\taskname` und – falls der Typ der Abschlussarbeit angegeben wurde – noch aus `\tasktext` und `\thesis` zusammensetzt. Der Parameter `headline` kann genutzt werden, um diese automatisch generierte Überschrift anzupassen.

Zu Beginn der Aufgabenstellung erscheint eine Tabelle mit den angegebenen Informationen zum Autor respektive zu den Autoren der Abschlussarbeit. Zwingend anzugeben sind dafür lediglich der Name des oder der Verfasser (`\author`) sowie der Titel der Arbeit (`\title`), welcher am Ende der Tabelle in fetter Schrift aufgeführt wird. Optional werden noch die Felder für den Studiengang (`\course`), die Fachrichtung (`\discipline`) sowie die Matrikelnummer (`\matriculationnumber`) und das Immatrikulationsjahr (`\matriculationyear`) ausgegeben, wobei nicht angegebene Felder bei der Ausgabe ignoriert werden. Der eigentliche Inhalt der Umgebung – sprich die Aufgabenstellung selbst – wird nach dem generierten Kopf ausgegeben.

Dem Inhalt der Aufgabenstellung folgt eine zeilenweise Auflistung des oder der definierten Gutachter beziehungsweise Prüfer (`\referee`) sowie Betreuer (`\supervisor`). Dabei wird vor dem jeweiligen Namen der dazugehörige Bezeichner (`\refereename`, `\refereeothername` respektive `\supervisorname`, `\supervisorothername`) gesetzt. Dies ist das voreingestellte Verhalten und kann über die Wahl des Parameters `style=table` aktiviert werden. Wird hingegen der Parameter `style=inline` gesetzt, so erfolgt die Ausgabe von mehreren Prüfern und Betreuern in einer Zeile. Die Bezeichner sind problemlos anpassbar, siehe dazu Unterkapitel 2.4. Danach erscheinen das Ausgabedatum (`\issuedate`) sowie der verpflichtende Abgabetermin (`\duedate`). Zum Schluss wird die Unterschriftzeile für den Prüfungsausschussvorsitzenden (`\chairman`) und den betreuenden Hochschullehrer (`\professor`) gesetzt. Für genannte Personen werden unter dem Namen selbst die Bezeichner ausgegeben (`\chairmanname` und `\professorname`).

v2.05

```
\taskform[<Parameterliste>]{<Ziele>}{<Schwerpunkte>}
```

Zusätzlich zur der frei gestaltbaren Umgebung `task` zur Erstellung einer Aufgabenstellung wird ein separater Befehl für eine standardisierte Ausgabe zur Verfügung gestellt. Dieser strukturiert die Aufgabenstellung in die zwei Bereiche *Ziele* und *Schwerpunkte* der Arbeit mit dazugehörigen Überschriften (`\objectivesname`, `\focusname`).



Im optionalen Argument können alle Parameter der Umgebung `task` verwendet werden. Im ersten obligatorischen Argument sollte ein Text mit einer kurzen thematischen Einordnung und dem eigentlichen Ziel der Arbeit erscheinen, im zweiten Argument sollen die thematischen Schwerpunkte in Stichpunkten benannt werden. Der Inhalt des zweiten notwendigen Argumentes wird in einer `itemize`-Umgebung gesetzt. Deshalb *muss* jedem Stichpunkt `\item` vorangestellt werden.

**Beispiel:** Die empfohlene Verwendung des Befehls `\taskform` ist wie folgt:

```
\taskform{%
  Motivation der Arbeit im ersten Absatz. . .

  Ziele der Arbeit im zweiten Absatz. . .
}%
\item Schwerpunkt 1
\item Schwerpunkt 2
}
```

Hierzu sei auch auf das Minimalbeispiel in Unterkapitel 6.4 verwiesen.

```
\chairman{<Prüfungsausschussvorsitzender>}
```

Wird dieses Feld genutzt, wird neben dem betreuenden Hochschullehrer (`\professor`) auch der Vorsitzende des Prüfungsausschusses am Ende der Aufgabenstellung aufgeführt. Dies wird zumeist für Abschlussarbeiten wie beispielsweise Master-Arbeit oder Diplomarbeit benötigt.

```
\issuedate{<Ausgabedatum>}
\duedate{<Abgabetermin>}
```

Mit diesen beiden Befehlen sollte das Datum der Ausgabe der Aufgabenstellung sowie der spätest mögliche Abgabetermin angegeben werden. Wurde das Paket `isodate` oder `datetime2` geladen, wird die damit eingestellte Ausgabeformatierung des Datums mit `\printdate` beziehungsweise `\DTMDate` für `\duedate` und `\issuedate` verwendet.

## 4.2 Gutachten für wissenschaftliche Arbeiten

```
\begin{evaluation}[<Überschrift> || <Parameterliste>] (tudpage in Abschnitt 2.2.3)
. . .
\end{evaluation}
headline=<Überschrift> (Parameter)
grade=<Note> (Parameter)
```

Diese Umgebung wird für das Erstellen eines Gutachtens einer wissenschaftlichen Arbeit bereitgestellt. Auch diese unterstützt alle Parameter, welche für die Umgebung `tudpage` in Abschnitt 2.2.3 auf Seite 32 beschrieben wurden.

Für ein Gutachten wird gewöhnlich eine Überschrift aus `\evaluationname` und – falls der Abschlussarbeitstyp angegeben wurde – `\evaluationtext` sowie `\thesis` generiert. Diese automatisch generierte Überschrift kann mit dem Parameter `headline` ersetzt werden. Am Ende des Gutachtens wird die mit `grade` gegebene Note in fetter Schrift ausgezeichnet.

Am Anfang der `evaluation`-Umgebung wird die gleiche Tabelle mit Autorenangaben ausgegeben, wie dies bei der `task`-Umgebung der Fall ist. Nach dem Tabellenkopf folgt auch hier der eigentliche Inhalt, sprich das Gutachten der Abschlussarbeit. Abgeschlossen wird die Umgebung

mit der gegebenen Note – welche innerhalb von `\gradetext` ausgegeben wird – sowie der Orts- und Datumsangabe (`\place`, `\date`) und der darauffolgenden Unterschriftzeile für den oder die Gutachter (`\referee`), welche wiederum mit den entsprechenden sprachabhängigen Bezeichner (`\refereename`, `\refereeothername`) ergänzt werden.

```
\evaluationform[<Parameterliste>]{<Aufgabe>}{<Inhalt>}{<Bewertung>}{<Note>}
```

Neben der individuell nutzbaren Umgebung `evaluation` wird ein separater Befehl zur Erstellung eines standardisierten Gutachtens bereitgestellt. Dieser strukturiert die Ausgabe in die vier Bereiche *Aufgabe*, *Inhalt*, *Bewertung* und *Note* und versieht diese jeweils mit der dazugehörigen Überschrift beziehungsweise Textausgabe (`\taskname`, `\contentname`, `\assessmentname` und `\gradetext`). Das optionale Argument unterstützt alle Parameter der `evaluation`-Umgebung.

**Beispiel:** Die empfohlene Verwendung des Befehls `\evaluationform` ist wie folgt:

```
\evaluationform{%
  Kurzbeschreibung der Aufgabenstellung. . .
}{%
  Zusammenfassung von Inhalt und Struktur. . .
}{%
  Bewertung der schriftlichen Abschlussarbeit. . .
}{%
  Zahl (Note)
}
```

Hierzu sei auch auf das Minimalbeispiel in Unterkapitel 6.5 verwiesen.

```
\grade{<Note>}
```

Neben der Angabe der Note für ein Gutachten über den Parameter `grade` der Umgebung `evaluation` kann dafür auch dieser global wirkende Befehl verwendet werden.

## 4.3 Aushang

```
\begin{notice}[<Überschrift> || <Parameterliste>] (tudpage in Abschnitt 2.2.3)
. . .
\end{notice}
headline=<Überschrift> (Parameter)
```

Für das Anfertigen eines Aushangs kann diese Umgebung verwendet werden. Diese basiert abermals auf der Umgebung `tudpage` und unterstützt alle deren Parameter.

Wurde ein Datum angegeben, wird dieses standardmäßig rechtsbündig oberhalb des Textbereiches angezeigt (*siehe* `cdhead`). Danach erfolgt die Ausgabe der Überschrift, welche für gewöhnlich dem Inhalt von `\noticename` entspricht und mit `headline` geändert werden kann. Nach der Überschrift folgt der Inhalt der Umgebung. Wurde mit `\contactperson` ein oder mehrere Ansprechpartner angegeben, werden diese Informationen am Ende der Umgebung gesetzt.

```
\noticeform[<Parameterliste>]{<Inhalt>}{<Schwerpunkte>}
```

Auch für diese Umgebung gibt es einen Befehl für eine normierte Form. Diese soll vor allem Verwendung für den Aushang studentischer Arbeitsthemen finden. Für das optionale Argument können sämtliche Parameter verwendet werden, die auch die `notice`-Umgebung unterstützt.

Das erste obligatorische Argument wird für eine kurze Inhaltsbeschreibung verwendet. Neben dem Text sollte hier wenn möglich eine thematisch passende Abbildung eingefügt werden (`\includegraphics`). Das zweite Argument wird – wie schon bei `\taskform` – dazu verwendet, einige Schwerpunkte aufzuzählen. Auch hier kommt nach der gliedernden Überschrift (`\focusname`) eine `itemize`-Umgebung zum Einsatz, allen Schwerpunkten muss ein `\item` vorangestellt werden.

**Beispiel:** Die empfohlene Verwendung des Befehls `\noticeform` ist wie folgt:

```
\noticeform{%
  Kurzbeschreibung des Inhaltes der studentischen Arbeit. . .

  Bild (optional), einzubinden mit:
  \includegraphics[<Einstellungen>]{<Datei>}
}%
\item Schwerpunkt 1
\item Schwerpunkt 2
}
```

Hierzu sei auch auf das Minimalbeispiel in Unterkapitel 6.6 verwiesen.

v2.02  
v2.05

```
\contactperson{<Kontaktperson(en)>}
\office{<Dienststiz>}
\telephone{<Telefonnummer>}
\telefax{<Telefaxnummer>}
```

Am Ende eines Aushangs können mit `\contactperson` Kontaktinformationen für eine oder mehrere Ansprechpartner angegeben werden. Soll mehr als eine Kontaktperson genannt werden, so müssen diese innerhalb des Befehls `\contactperson` mit dem Befehl `\and` getrennt werden. Für jede Person kann innerhalb des Argumentes von `\contactperson` der Dienststiz (`\office`), die dienstliche Telefon- (`\telephone`) sowie -faxnummer (`\telefax`) und die geschäftliche E-Mail-Adresse (`\emailaddress`) angegeben werden.

## 4.4 Zusätzliche sprachabhängige Bezeichner

Für das Paket `tudscrsupervisor` werden für die zusätzlichen Befehle und Umgebungen weitere Bezeichner definiert. Für eine etwaige Anpassung dieser sei auf Unterkapitel 2.4 verwiesen.

```
\taskname (Bezeichner)
\tasktext (Bezeichner)
```

Die Bezeichnung der Aufgabenstellung selbst ist in `\taskname` enthalten. Für die Generierung einer Überschrift wird dieser verwendet. Wurde außerdem mit `\thesis` oder `\subject` der Typ der Abschlussarbeit<sup>38</sup> angegeben, wird die Überschrift zusammen mit dem Bezeichner `\tasktext` um die Typisierung erweitert. Falls gewünscht, kann die automatisch generierte Überschrift mit dem Parameter `headline` der Umgebung `task` überschrieben werden.

Bezeichner	Deutsch	Englisch
<code>\taskname</code>	Aufgabenstellung	Task
<code>\tasktext</code>	für die Anfertigung einer	for the preparation of a

<sup>38</sup>`subjectthesis` oder spezieller Wert aus Tabelle 2.1

`\namesname` (Bezeichner)`\titlename` (Bezeichner)

Diese beiden Bezeichner werden in der Tabelle mit den Autoreninformationen zu Beginn der Aufgabenstellung verwendet.

Bezeichner	Deutsch	Englisch
<code>\namesname</code>	Name	Name
<code>\titlename</code>	Titel	Title

`\issuedatetext` (Bezeichner)`\duedatetext` (Bezeichner)

Am Ende der Aufgabenstellung wird nach dem oder der Betreuer das Ausgabedatum und der Abgabetermin (`\issuedate`, `\duedate`) der Abschlussarbeit mit folgenden Bezeichner erläutert.

Bezeichner	Deutsch	Englisch
<code>\issuedatetext</code>	Ausgehändigt am	Issued on
<code>\duedatetext</code>	Einzureichen am	Due date for submission

`\chairmanname` (Bezeichner)

Wurde der Prüfungsausschussvorsitzende (`\chairman`) angegeben, erfolgt unter dem Namen selbst die Ausgabe des Bezeichners.

Bezeichner	Deutsch	Englisch
<code>\chairmanname</code>	Prüfungsausschussvorsitzender	Chairman of the Audit Committee

`\focusname` (Bezeichner)`\objectivesname` (Bezeichner)

Die Standardformen für Aufgabenstellung (`\taskform`) respektive Aushang (`\noticeform`) nutzen für die gesetzten Überschriften diese Bezeichner.

Bezeichner	Deutsch	Englisch
<code>\focusname</code>	Schwerpunkte der Arbeit	Focus of work
<code>\objectivesname</code>	Ziele der Arbeit	Objectives of work

`\evaluationname` (Bezeichner)`\evaluationtext` (Bezeichner)

Die Bezeichnung des Gutachten selbst ist in `\evaluationname` enthalten. Für die Generierung der Überschrift wird der Bezeichner `\evaluationtext` sowie der mit `\thesis` oder gegebenenfalls mit `\subject` gegebenen Typ der Abschlussarbeit verwendet. Diese automatisch generierte Überschrift kann mit dem Parameter `headline` der Umgebung `evaluation` durch den Anwender überschrieben werden.

Bezeichner	Deutsch	Englisch
<code>\evaluationname</code>	Gutachten	Evaluation
<code>\evaluationtext</code>	für die	for the

\contentname (Bezeichner)  
 \assessmentname (Bezeichner)

Bei der standardisierten Form des Gutachtens (\evaluationform) werden die darin – für eine strukturierte Gliederung – erzeugten Überschriften mit den Bezeichnern \taskname, \contentname und \assessmentname gesetzt.

Bezeichner	Deutsch	Englisch
\taskname	Aufgabenstellung	Task
\contentname	Inhalt	Content
\assessmentname	Bewertung	Assessment

\gradetext (Bezeichner)

Wird für das Gutachten einer wissenschaftlichen Arbeit die erzielte Note entweder mit dem Befehl \grade{<Note>} oder alternativ dazu mit dem Parameter grade=<Note> der Umgebung evaluation angegeben, so wird diese innerhalb von \gradetext verwendet.

Bezeichner	Deutsch	Englisch
\gradetext	Die Arbeit wird mit der Note <Note> bewertet.	The thesis is evaluated with a grade of <Note>.

v2.02

\noticename (Bezeichner)  
 \contactpersonname (Bezeichner)

Die Bezeichnung des Aushangs selbst ist in \noticename enthalten. Für die Generierung einer Überschrift wird dieser verwendet. Falls gewünscht, kann diese mit dem Parameter headline der Umgebung notice überschrieben werden. Wurde eine Kontaktperson mit dem Befehl \contactperson angegeben, wird als Überschrift der Kontaktdaten der Bezeichner \contactpersonname verwendet.

Bezeichner	Deutsch	Englisch
\noticename	Aushang	Notice
\contactpersonname	Ansprechpartner	Counterpart

## Kapitel 5

# Zusätzliche Pakete im TUD-Script-Bundle

Zusätzlich zu den bisher im Anwenderhandbuch vorgestellten Klassen und Paketen werden im TUD-Script-Bundle weitere Pakete bereitgestellt. Diese sind nicht zwingend an die Verwendung einer der TUD-Script-Klassen angewiesen sondern können prinzipiell mit jeder  $\text{\LaTeX} 2_{\epsilon}$ -Klasse genutzt werden.

## 5.1 Das Paket tudscrcolor – Farben im Corporate Design

Zur Verwendung der Farben des Corporate Designs wird das Paket `tudscrcolor` genutzt. Falls dieses nicht in der Präambel geladen wird – um beispielsweise zusätzliche Optionen aufzurufen – binden die TUD-Script-Klassen dieses automatisch ein.

Für das Corporate Design sind mehrere Farben vorgesehen. Die prägnanteste aller ist die Hausfarbe HKS41, danach folgen die Farben für Auszeichnungen der ersten (HKS44) und der zweiten Kategorie (HKS36, HKS33, HKS57, HKS65) sowie eine Ausnahmefarbe (HKS07). Diese Farben dürfen sowohl in ihrer Grundform als auch in helleren Tönen mit einer Abstufung in 10 %-Schritten verwendet werden. Das ohnehin verwendete Paket `xcolor` stellt genau diese Funktionalität zur Verfügung. Jede der Farben kann sowohl mit `HKS<Zahl>` als auch über ein Pseudonym `cd<Farbe>` genutzt werden.

**Beispiel:** Die Grundfarbe HKS44 soll in der auf 20% reduzierten, helleren Abstufung genutzt werden. Innerhalb eines Befehls, der als Argument eine gültige Farbe erwartet, muss lediglich `HKS44!20` angegeben werden. Dies wird hier exemplarisch mit der folgenden `Box \colorbox{HKS44!20}{Box}` demonstriert.

Bei der farbigen Gestaltung des Corporate Designs (`cd=color`) ist der Hintergrund von Umschlagseite, Titel sowie Teilen in HKS41 und die Schrift auf selbigen in `HKS41!30` gehalten. Der Hintergrund von Kapitelseiten erscheint in `HKS41!10`, die Schrift in HKS41. Bei geringerem Farbeinsatz werden lediglich die Schriften der Gliederungsseiten auf HKS41 gesetzt.

Sollen bestimmte Optionen an das Paket `xcolor` weitergereicht werden, gibt es dafür zwei Möglichkeiten. Diese kann entweder vor dem Laden der Klasse direkt an `xcolor` übergeben werden<sup>39</sup> oder es wird `tudscrcolor` mit der entsprechenden Option geladen.<sup>40</sup>

### 5.1.1 Generelle Farbdefinitionen

#### Primäre Hausfarbe

HKS41 (`cddarkblue`) (Farbe)



#### Sekundäre Hausfarbe (Geschäftsausstattung)

HKS92 (`cdgray`) (Farbe)

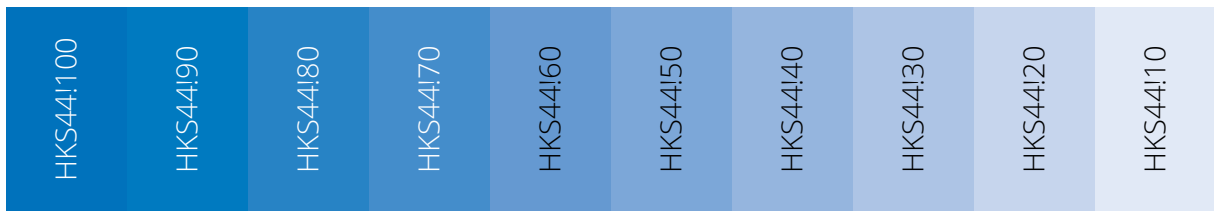


<sup>39</sup>`\PassOptionsToPackage{<Paketoptionen>}{xcolor}` vor dem Laden der genutzten Dokumentklasse mit `\documentclass[<Klassenoptionen>]{<Klasse>}`

<sup>40</sup>`\usepackage[<Paketoptionen>]{tudscrcolor}`; `tudscrcolor` reicht `<Paketoptionen>` an `xcolor` weiter

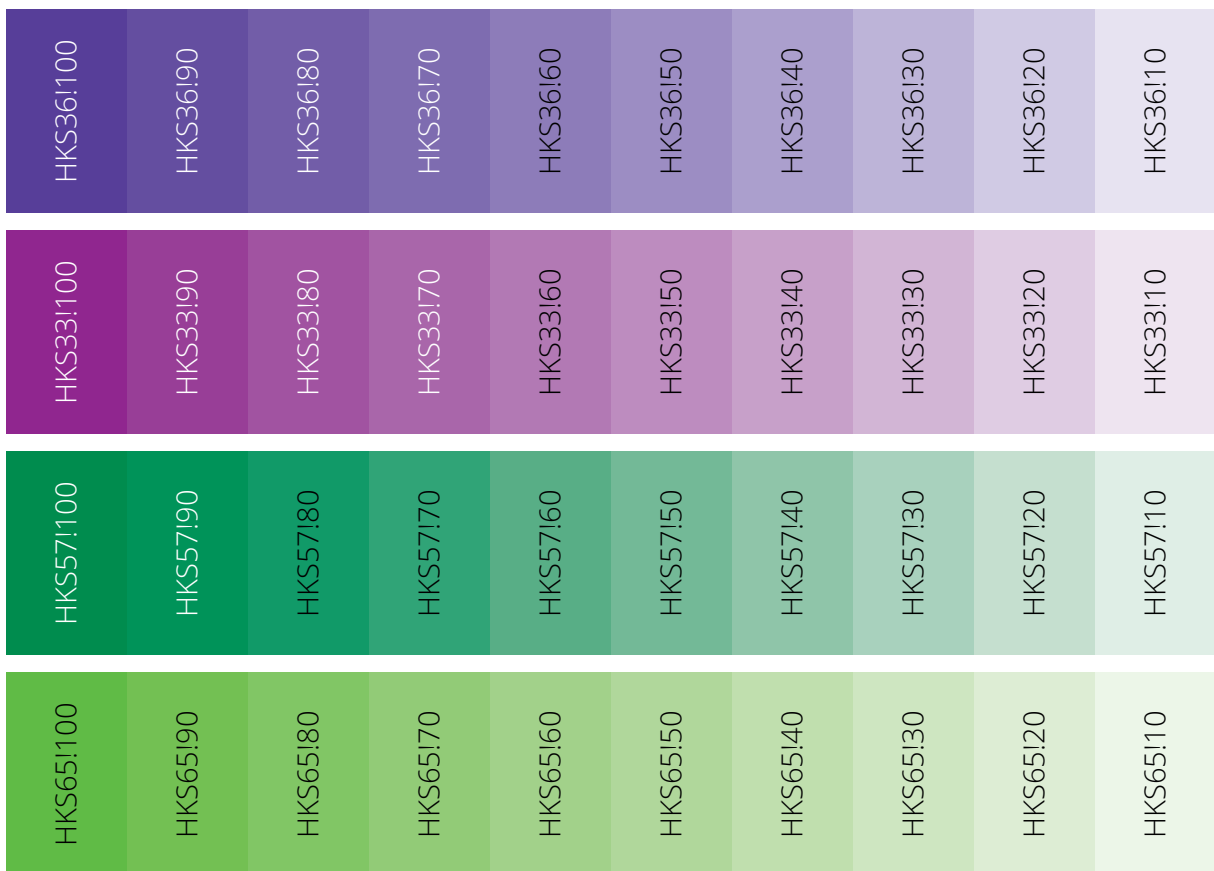
## Auszeichnungsfarbe 1. Kategorie

HKS44 (cdblue) (Farbe)



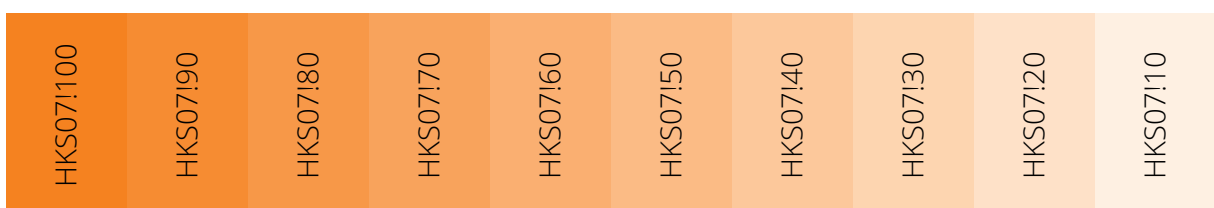
## Auszeichnungsfarbe 2. Kategorie

HKS36 (cdindigo) (Farbe)  
HKS33 (cdpurple) (Farbe)  
HKS57 (cddarkgreen) (Farbe)  
HKS65 (cdgreen) (Farbe)



## Ausnahmefarbe

HKS07 (cdorange) (Farbe)





## 5.1.2 Zusätzliche Farbdefinitionen

Das Paket **tudscrcolor** definiert im Normalfall lediglich die zuvor beschriebenen Grundfarben HKS41, HKS92, HKS44, HKS36, HKS33, HKS57, HKS65 sowie HKS07. Alle anderen farblichen Abstufungen können mit den beschriebenen Möglichkeiten des Paketes **xcolor** generiert werden.

`oldcolors`

In den letzten Jahren sind viele verschiedene Klassen und Pakete für das Corporate Design der Technischen Universität Dresden entstanden. Innerhalb dieser existieren abweichende Farbdefinitionen. Um eine Migration von den benannten Klassen und Paketen auf TUD-Script zu ermöglichen, existiert die Paketoption `oldcolors`. Wird diese genutzt, so werden zusätzliche Farben nach dem Schema `HKS41K<Zahl>` und `HKS41-<Zahl>` definiert, wobei der hinten angestellte Zahlenwert aus der 10er-Reihe kommen muss.

## 5.1.3 Umstellung des Farbmodells

Normalerweise verwendet **tudscrcolor** das CMYK-Farbmodell. Außerdem wird weiterhin noch der RGB-Farbraum unterstützt. Eine Umschaltung des Farbmodells ist beispielsweise für gewisse Funktionen des Paketes **tikz** notwendig.

`RGB`

Mit dieser Option werden bereits beim Laden des Paketes **tudscrcolor** die Farben nicht nach dem CMYK-Farbmodell sondern im RGB-Farbraum global definiert.

`\setcdcolors{<Farbmodell>}`

Mit diesem Befehl kann innerhalb des Dokumentes das verwendete Farbmodell angepasst werden. Damit ist es möglich, lokal innerhalb einer Umgebung den Farbmodus zu ändern und so nur in bestimmten Situationen beispielsweise aus dem CMYK-Farbmodell in den RGB-Farbraum zu wechseln. Unterstützte Werte für `<Farbmodell>` sind CMYK und RGB beziehungsweise `rgb`.

Achtung!

*Die Darstellung der Farben kann im jeweiligen Farbmodus (CMYK oder RGB) je nach verwendeter Bildschirm-, Drucker- und Softwarekonfiguration verschieden ausfallen. Die Farbwerte entstammen dem Handbuch zum Corporate Design und sind lediglich Näherungswerte. Abweichungen vom gedruckten HKS-Farbbregister und selbst ermittelten Werten sind technisch nicht zu vermeiden.*

## 5.2 Das Paket tudscrfonts – Schriften im Corporate Design

v2.02

Dieses Paket stellt die Schriften des Corporate Designs für  $\text{\LaTeX} 2_{\epsilon}$ -Klassen bereit, welche *nicht* zum TUD-Script-Bundle gehören. Das Paket unterstützt einen Großteil der normalerweise für die TUD-Script-Klassen bereitgestellten Optionen und Befehle für die Schriftauswahl. Um Dopplungen in der Dokumentation zu vermeiden, wird an dieser Stelle auf eine abermalige Erläuterung der im Paket **tudscrfonts** verfügbaren Optionen und Befehle verzichtet. Diese werden im Folgenden lediglich noch einmal genannt, die dazugehörigen Erläuterungen sind in Unterkapitel 2.1 auf Seite 15 zu finden.

Die nutzbaren Paketoptionen sind für den Fließtext `cdfont` – ohne die Einstellungsmöglichkeiten für den Querbalken des Corporate Designs (`cdhead`) – und für die mathematischen Schriften `cdmath` sowie `s1antedgreek`. Für die Auswahl der Schreibmaschinenschriften ist `ttfont` verfügbar. Weiterhin wird die Option `relspacing` bereitgestellt. Alle genannten Optionen können entweder direkt als Klassenoption oder als Paketoptionen im optionalen Argument

von `\usepackage[<Paketoptionen>]{tudscrfonts}` angegeben werden. Zusätzlich ist nach dem Laden des Paketes die Optionenwahl mit `\TUDoption` respektive `\TUDoptions` möglich.

v2.06

Die in Unterkapitel 2.1 beschriebenen Textschalter und -kommandos zur expliziten Auswahl einzelner Schnitte der Hausschrift des Corporate Designs sowie die Befehle für griechische Buchstaben werden ebenso bereitgestellt. Soll nicht **Open Sans** sondern die alten Schriftfamilien **Univers** und **DIN Bold** genutzt werden, ist eine Installation dieser notwendig. Hinweise hierzu sind unter Unterkapitel A.4 zu finden. Um diese zu aktivieren, sei auf `tudscrver=2.05` und `cdo1dfont` verwiesen.

v2.04  
v2.05

Ursprünglich war das Paket `tudscrfonts` für die Verwendung zusammen mit einer der Klassen `tudbook`, `tudbeamer`, `tudletter`, `tudfax`, `tudhaus`, `tudform` und seit der Version v2.04 auch `tudmathposter` sowie `tudposter` vorgesehen. Allerdings traten bei der Verwendung des Paketes mit einer dieser Klassen einige kleinere Unzulänglichkeiten auf. Deshalb wird seit der Version v2.05 empfohlen, für diese Klassen das Paket `fix-tudscrfonts` zu verwenden.

### 5.3 Das Paket `fix-tudscrfonts` – Schriftkompatibilität

v2.05

Dieses Paket ist für die alleinige Verwendung mit einer der folgenden Klassen vorgesehen:

- `tudbook`
- `tudbeamer`
- `tudletter`
- `tudfax`
- `tudhaus`
- `tudform`
- `tudmathposter`
- `tudposter`

v2.06

Für das Corporate Design der Technischen Universität Dresden wird vom TUD-Script-Bundle standardmäßig **Open Sans** als Schriftfamilie verwendet. Diese unterscheidet sich stark von den ursprünglich durch die gerade genannten Klassen genutzten Schriftfamilien. Wird nun das Paket `fix-tudscrfonts` zusammen mit einer dieser Klassen verwendet, hat dies den Vorteil, dass auch bei diesen die aktuelle Hausschrift verwendet wird und der stark verbesserte Mathematiksatz zum Tragen kommen. *In diesem Fall kann sich das Ausgabeergebnis im Vergleich zu der Varianten mit den alten Schriften ändern. Zum Setzen von Dokumenten mit den alten Schriftfamilien **Univers** und **DIN Bold** müssen diese installiert werden. Hinweise hierzu sind unter Unterkapitel A.4 zu finden. Um diese zu aktivieren, sei auf `tudscrver=2.05` und `cdo1dfont` verwiesen.*

Achtung!

Um alle notwendigen Einstellung korrekt und ohne unnötige Warnungen vornehmen zu können, muss das Paket `fix-tudscrfonts` bereits vor der Dokumentklasse geladen werden, wobei die gleichen Paketoptionen wie für das Paket `tudscrfonts` verwendet werden können:

```
\RequirePackage[<Paketoptionen>]{fix-tudscrfonts}
\documentclass[<Klassenoptionen>]{<Dokumentklasse>}
. . .
\begin{document}
. . .
\end{document}
```

Dabei wird spätestens zum Ende der Präambel das Paket `tudscrfonts` geladen. Alternativ kann dies auch durch den Benutzer in der Dokumentpräambel erfolgen.

## 5.4 Das Paket mathswap

Die Verwendung von Dezimal- und Tausendertrennzeichen im mathematischen Satz sind regional sehr unterschiedlich. In den meisten englischsprachigen Ländern wird der Punkt als Dezimaltrennzeichen und das Komma zur Zifferngruppierung verwendet, im restlichen Europa wird dies genau entgegengesetzt praktiziert. Dieses Paket soll dazu dienen, beliebige formatierte Zahlen in ihrer Ausgabe anzupassen. Dafür werden die Zeichen Punkt (.) und Komma (,) als aktive Zeichen im Mathematikmodus definiert.

Ähnliche Funktionalitäten werden bereits durch die Pakete **icomma** und **ziffer** bereitgestellt. Bei **icomma** muss jedoch beim Verfassen des Dokumentes durch den Autor beachtet werden, ob das verwendete Komma einem Dezimaltrennzeichen entspricht ( $t = 1,2$ ) oder einem normalen Komma im Mathematiksatz ( $z = f(x,y)$ ), wo ein gewisser Abstand nach dem Komma durchaus gewünscht ist. Das Paket **ziffer** liefert dafür die gewünschte Funktionalität,<sup>41</sup> ist allerdings etwas unflexibel, was den Umgang mit den Trennzeichen anbelangt. Als Alternative zu diesem Paket kann außerdem **ionumbers** verwendet werden.

Das Paket **mathswap** sorgt dafür, dass Trennzeichen direkt vor einer Ziffer erkannt und nach bestimmten Vorgaben ersetzt werden. Sollte sich jedoch zwischen Trennzeichen und Ziffer Leerraum befinden, wird dieser als solcher auch gesetzt. Für ein Beispiel zur Verwendung des Paketes sei auf das Tutorial **mathswap.pdf** in Kapitel 7 hingewiesen.

```
\commaswap{<Trennzeichen>}
\dotswap{<Trennzeichen>}
```

Die beiden Befehle `\commaswap` und `\dotswap` sind die zentrale Benutzerschnittstelle des Paketes. Das Makro `\commaswap` definiert das Trennzeichen oder den Inhalt, wodurch ein Komma ersetzt werden soll, auf welches direkt danach eine Ziffer folgt. Normalerweise setzt  $\LaTeX$  nach einem Komma im mathematischen Satz zusätzlich einen horizontalen Abstand. Bei der Ersetzung durch `\commaswap` entfällt dieser. Die Voreinstellung für `\commaswap` ist deshalb auf ein Komma (,) gesetzt. Mit dem Makro `\dotswap` kann definiert werden, wodurch der Punkt im mathematischen Satz ersetzt werden soll, wenn auf diesen direkt anschließend eine Ziffer folgt. Da der Punkt im deutschsprachigem Raum zur Gruppierung von Ziffern genutzt wird, ist hierfür standardmäßig ein halbes geschütztes Leerzeichen definiert (`\,.`).

v2.02

```
\mathswapon
\mathswapoff
```

Die Funktionalität von **mathswap** kann innerhalb des Dokumentes mit diesen beiden Befehlen an- und abgeschaltet werden. Beim Laden des Paketes ist es standardmäßig aktiviert.

## 5.5 Das Paket twocolfix

Der  $\LaTeX 2_{\epsilon}$ -Kernel enthält einen Fehler, der Kapitelüberschriften im zweispaltigen Layout höher setzt, als im einspaltigen. Der Fehler<sup>42</sup> ist zwar schon länger bekannt, allerdings bisher noch nicht in den  $\LaTeX 2_{\epsilon}$ -Kernel übernommen worden. Das Paket **twocolfix** behebt das Problem. Eine Integration dieses Bugfixes in KOMA-Script wurde bereits bei Markus Kohm angefragt, jedoch von ihm bis jetzt nicht weiter verfolgt<sup>43</sup>.

<sup>41</sup>kein Leerraum nach Komma, wenn direkt danach eine Ziffer folgt

<sup>42</sup><http://www.latex-project.org/cgi-bin/ltxbugs2html?pr=latex/3126>

<sup>43</sup><http://www.komascript.de/node/1681>

## Zukünftige Arbeiten

Diese Dinge sollen langfristig in das TUD-Script-Bundle eingearbeitet werden:

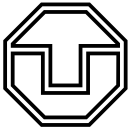
Es soll das Paket **tudscrletter** für Briefe im Corporate Design der Technischen Universität Dresden entstehen. Auch Klassen für Fax und Hausmitteilungen sollen dabei abfallen.

Mit **tudscrbeamer** soll ein Paket entstehen, mit dem sich  $\LaTeX$ -Beamer-Präsentationen im Stil des Corporate Designs der Technischen Universität Dresden mit den Einstellungsmöglichkeiten von TUD-Script erstellen lassen. Aktuell können hierfür die Pakete aus dem GitHub-Repository **tud-cd**<sup>44</sup> genutzt werden.

Außerdem ist ein Paket **tudscrlayout** vorstellbar, welches den durch das Corporate Design vorgegebene Satzspiegel aktiviert ohne den Seitenstil selbst zu verwenden, um beispielsweise bereits mit dem Kopf der Technischen Universität Dresden bedrucktes Papier nutzen zu können. Ebenfalls wäre es denkbar, für andere Klassen die `tudheadings`-Seitenstile verfügbar zu machen ohne dabei den Satzspiegel des Corporate Designs umzusetzen. Dies hat jedoch nur geringe Priorität.

---

<sup>44</sup><https://github.com/tud-cd/tud-cd/>



## Teil II

# Ergänzungen und Hinweise

Für die Verwendung des TUD-Script-Bundles ist es nicht notwendig, diesen Teil zu lesen. In Kapitel 6 werden insbesondere für  $\LaTeX$ -Neulinge sowie neue Anwender des TUD-Script-Bundles mehrere einfache Beispiele gezeigt. Zusätzlich sind in Kapitel 7 umfangreichere Tutorials für das Erstellen von wissenschaftlichen Arbeiten zu finden. Die darin gegebenen Empfehlungen sind nicht auf TUD-Script beschränkt sondern lassen sich auch mit anderen  $\LaTeX$ -Klassen umsetzen. In Kapitel 8 werden Einsteigern – und versierten  $\LaTeX$ -Nutzern – meiner Meinung nach empfehlenswerte Pakete kurz vorgestellt.

Anwendungshinweise sowie der eine oder andere allgemeine Hinweis bei der Verwendung von  $\LaTeX 2_{\epsilon}$  wird in Kapitel 9 gegeben. Dabei sind diese durchaus für die Verwendung sowohl des TUD-Script-Bundles als auch anderer  $\LaTeX$ -Klassen interessant. Für Anregungen, Hinweise, Ratschläge oder Empfehlungen zu weiteren Pakete sowie Tipps bin ich jederzeit empfänglich.

## Kapitel 6

# Minimalbeispiele

Dieses Kapitel soll den Einstieg und den ersten Umgang mit TUD-Script erleichtern. Dafür werden einige Minimalbeispiele gegeben, die einzelne Funktionalitäten darstellen. Diese sind so reduziert ausgeführt, dass sie sich dem Anwender direkt erschließen sollten.

## 6.1 Dokument

Hier wird gezeigt, wie die Präambel eines minimalen  $\text{\LaTeX} 2_{\epsilon}$ -Dokumentes gestaltet werden sollte. Dieser Ausschnitt kann prinzipiell als Grundlage für ein neu zu erstellendes Dokument verwendet werden. Lediglich das Einbinden des Paketes **blindtext** mit `\usepackage{blindtext}` und die Verwendung des daraus stammenden Befehls `\blinddocument` können weggelassen werden.

```
\documentclass[ngerman]{tudscrreprt}
\iftutex
  \usepackage{fontspec}
\else
  \usepackage[T1]{fontenc}
  \usepackage[ngerman=ngerman-x-latest]{hyphsubst}
\fi
\usepackage{babel}
\usepackage[math]{blindtext}
\begin{document}
\faculty{Juristische Fakultät}
\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}
\chair{Lehrstuhl für Kriminalprognose}
\date{18.02.2015}
\author{Mickey Mouse}
\title{Der Geldspeicher in Entenhausen}
\maketitle
\blinddocument
\end{document}
```

# Der Geldspeicher in Entenhausen

Mickey Mouse

18.02.2015



# 1 Überschrift auf Ebene 0 (chapter)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld.  $\sin^2(\alpha) + \cos^2(\beta) = 1$ . Der Text gibt lediglich den Grauwert der Schrift an  $E = mc^2$ . Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen.  $\sqrt[3]{a} \cdot \sqrt[3]{b} = \sqrt[3]{ab}$ . An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft.  $\frac{\sqrt[3]{a}}{\sqrt[3]{b}} = \sqrt[3]{\frac{a}{b}}$ . Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein.  $a\sqrt[3]{b} = \sqrt[3]{a^3b}$ . Er muss keinen Sinn ergeben, sollte aber lesbar sein.  $d\Omega = \sin\theta d\theta d\varphi$ . Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

## 1.1 Überschrift auf Ebene 1 (section)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld.  $\sin^2(\alpha) + \cos^2(\beta) = 1$ . Der Text gibt lediglich den Grauwert der Schrift an  $E = mc^2$ . Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen.  $\sqrt[3]{a} \cdot \sqrt[3]{b} = \sqrt[3]{ab}$ . An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft.  $\frac{\sqrt[3]{a}}{\sqrt[3]{b}} = \sqrt[3]{\frac{a}{b}}$ . Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein.  $a\sqrt[3]{b} = \sqrt[3]{a^3b}$ . Er muss keinen Sinn ergeben, sollte aber lesbar sein.  $d\Omega = \sin\theta d\theta d\varphi$ . Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

### 1.1.1 Überschrift auf Ebene 2 (subsection)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld.  $\sin^2(\alpha) + \cos^2(\beta) = 1$ . Der Text gibt lediglich den Grauwert der Schrift an  $E = mc^2$ . Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen.  $\sqrt[3]{a} \cdot \sqrt[3]{b} = \sqrt[3]{ab}$ . An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft.  $\frac{\sqrt[3]{a}}{\sqrt[3]{b}} = \sqrt[3]{\frac{a}{b}}$ . Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein.  $a\sqrt[3]{b} = \sqrt[3]{a^3b}$ . Er muss keinen Sinn ergeben, sollte aber lesbar sein.  $d\Omega = \sin\theta d\theta d\varphi$ . Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

### Überschrift auf Ebene 3 (subsubsection)

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld.  $\sin^2(\alpha) + \cos^2(\beta) = 1$ . Der Text gibt lediglich den Grauwert der Schrift an  $E = mc^2$ . Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburnn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen.  $\sqrt[3]{a} \cdot \sqrt[3]{b} = \sqrt[3]{ab}$ . An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft.  $\frac{\sqrt[3]{a}}{\sqrt[3]{b}} = \sqrt[3]{\frac{a}{b}}$ . Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein.  $a\sqrt[3]{b} = \sqrt[3]{a^3b}$ . Er muss keinen Sinn ergeben, sollte aber lesbar sein.  $d\Omega = \sin\theta d\theta d\varphi$ . Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

**Überschrift auf Ebene 4 (paragraph)** Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld.  $\sin^2(\alpha) + \cos^2(\beta) = 1$ . Der Text gibt lediglich den Grauwert der Schrift an  $E = mc^2$ . Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburnn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen.  $\sqrt[3]{a} \cdot \sqrt[3]{b} = \sqrt[3]{ab}$ . An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft.  $\frac{\sqrt[3]{a}}{\sqrt[3]{b}} = \sqrt[3]{\frac{a}{b}}$ . Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein.  $a\sqrt[3]{b} = \sqrt[3]{a^3b}$ . Er muss keinen Sinn ergeben, sollte aber lesbar sein.  $d\Omega = \sin\theta d\theta d\varphi$ . Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

## 1.2 Listen

### 1.2.1 Beispiel einer Liste (itemize)

- Erster Listenpunkt, Stufe 1
- Zweiter Listenpunkt, Stufe 1
- Dritter Listenpunkt, Stufe 1
- Vierter Listenpunkt, Stufe 1
- Fünfter Listenpunkt, Stufe 1

### Beispiel einer Liste (4\*itemize)

- Erster Listenpunkt, Stufe 1
  - Erster Listenpunkt, Stufe 2
    - \* Erster Listenpunkt, Stufe 3
      - Erster Listenpunkt, Stufe 4
      - Zweiter Listenpunkt, Stufe 4
    - \* Zweiter Listenpunkt, Stufe 3
  - Zweiter Listenpunkt, Stufe 2
- Zweiter Listenpunkt, Stufe 1

### 1.2.2 Beispiel einer Liste (enumerate)

1. Erster Listenpunkt, Stufe 1
2. Zweiter Listenpunkt, Stufe 1
3. Dritter Listenpunkt, Stufe 1
4. Vierter Listenpunkt, Stufe 1
5. Fünfter Listenpunkt, Stufe 1

### Beispiel einer Liste (4\*enumerate)

1. Erster Listenpunkt, Stufe 1
  - a) Erster Listenpunkt, Stufe 2
    - i. Erster Listenpunkt, Stufe 3
      - A. Erster Listenpunkt, Stufe 4
      - B. Zweiter Listenpunkt, Stufe 4
    - ii. Zweiter Listenpunkt, Stufe 3
  - b) Zweiter Listenpunkt, Stufe 2
2. Zweiter Listenpunkt, Stufe 1

### 1.2.3 Beispiel einer Liste (description)

- Erster Listenpunkt, Stufe 1
- Zweiter Listenpunkt, Stufe 1
- Dritter Listenpunkt, Stufe 1
- Vierter Listenpunkt, Stufe 1
- Fünfter Listenpunkt, Stufe 1

### Beispiel einer Liste (4\*description)

- Erster Listenpunkt, Stufe 1
- Erster Listenpunkt, Stufe 2
- Erster Listenpunkt, Stufe 3
- Erster Listenpunkt, Stufe 4
- Zweiter Listenpunkt, Stufe 4
- Zweiter Listenpunkt, Stufe 3
- Zweiter Listenpunkt, Stufe 2
- Zweiter Listenpunkt, Stufe 1

## 6.2 Dissertation

Eine Abschlussarbeit oder ähnliches könnte wie hier gezeigt begonnen werden.

```
\documentclass[%
  english,ngerman,%
  BCOR=6mm,cdgeometry=no,DIV=13%
]{tudscrreprt}
\iftutex
  \usepackage{fontspec}
\else
  \usepackage[T1]{fontenc}
  \usepackage[ngerman=ngerman-x-latest]{hyphsubst}
\fi
\usepackage{babel}
\usepackage{isodate}
\usepackage{blindtext}
\begin{document}
\faculty{Juristische Fakultät}
\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}
\chair{Lehrstuhl für Kriminalprognose}
\date{18.02.2015}
\author{Mickey Mouse}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\thesis{diss}
\graduation[Dr. jur.]{Doktor der Rechte}
\dateofbirth{2.1.1990}
\placeofbirth{Dresden}
\defensedate{15.03.2015}
\referee{Dagobert Duck \and Mac Moneysac}
\maketitle
\TUOption{abstract}{section,multiple}
\begin{abstract}[pagestyle=empty.tudheadings]
\blindtext
\nextabstract[english]
\blindtext
\end{abstract}
\end{document}
```

# Entwicklung eines optimalen Verfahrens zur Eroberung des Geldspeichers in Entenhausen

Mickey Mouse

Geboren am: 2. Januar 1990 in Dresden

## Dissertation

zur Erlangung des akademischen Grades

## Doktor der Rechte (Dr. jur.)

Erstgutachter

**Dagobert Duck**

Zweitgutachter

**Mac Moneysac**

Eingereicht am: 18. Februar 2015

Verteidigt am: 15. März 2015

## **Zusammenfassung**

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

## **Abstract**

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

### 6.3 Abschlussarbeit (kollaborativ)

Alle zusätzlichen Angaben außerhalb des Argumentes von `\author` werden für beide Autoren gleichermaßen übernommen.<sup>45</sup> Die Angaben innerhalb des Argumentes von `\author` werden den jeweiligen, mit `\and` getrennten Autoren zugeordnet.<sup>46</sup> Ohne die Verwendung von `\and` kann natürlich auch nur ein Autor aufgeführt werden. Außerdem sei auf die Verwendung von `\subject` anstelle von `\thesis` mit einem speziellen Wert aus Tabelle 2.1 hingewiesen.

```
\documentclass[ngerman,BCOR=4mm]{tudscrreprt}
\iftutex
  \usepackage{fontspec}
\else
  \usepackage[T1]{fontenc}
  \usepackage[ngerman=ngerman-x-latest]{hyphsubst}
\fi
\usepackage{babel}
\usepackage{isodate}
\begin{document}
\faculty{Juristische Fakultät}
\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}
\chair{Lehrstuhl für Kriminalprognose}
\date{18.02.2015}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\subject{master}
\graduation[M.Sc.]{Master of Science}
\author{%
  Mickey Mouse%
  \matriculationnumber{12345678}%
  \dateofbirth{2.1.1990}%
  \placeofbirth{Dresden}%
\and%
  Donald Duck%
  \matriculationnumber{87654321}%
  \dateofbirth{1.2.1990}%
  \placeofbirth{Berlin}%
}
\matriculationyear{2010}
\supervisor{Dagobert Duck \and Mac Moneysac}
\professor{Prof. Dr. Kater Karlo}
\maketitle
\end{document}
```

---

<sup>45</sup>In diesem Beispiel `\matriculationyear`

<sup>46</sup>In diesem Beispiel `\matriculationnumber`, `\dateofbirth` und `\placeofbirth`



**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**

---

Juristische Fakultät Fachrichtung Strafrecht, Institut für Kriminologie, Lehrstuhl für Kriminalprognose

---

Master-Arbeit

# Entwicklung eines optimalen Verfahrens zur Eroberung des Geldspeichers in Entenhausen

**Mickey Mouse**

Geboren am: 2. Januar 1990 in Dresden  
Matrikelnummer: 12345678  
Immatrikulationsjahr: 2010

**Donald Duck**

Geboren am: 1. Februar 1990 in Berlin  
Matrikelnummer: 87654321  
Immatrikulationsjahr: 2010

18. Februar 2015

Betreuer

**Dagobert Duck  
Mac Moneysac**

Betreuender Hochschullehrer  
**Prof. Dr. Kater Karlo**



## 6.4 Aufgabenstellung (kollaborativ)

Eine Aufgabenstellung für eine wissenschaftliche Arbeit ist mithilfe der Umgebung task oder dem Befehl \taskform aus dem Paket tudscrsupervisor folgendermaßen dargestellt werden.

```
\documentclass[ngerman]{tudscrrcpt}
\iftutex
  \usepackage{fontspec}
\else
  \usepackage[T1]{fontenc}
  \usepackage[ngerman=ngerman-x-latest]{hyphsubst}
\fi
\usepackage{babel}
\usepackage{isodate}
\usepackage{tudscrsupervisor}
\usepackage{enumitem}\setlist{noitemsep}
\begin{document}
\faculty{Juristische Fakultät}\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}\chair{Lehrstuhl für Kriminalprognose}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\thesis{master}\graduation[M.Sc.]{Master of Science}
\author{%
  Mickey Mouse\matriculationnumber{12345678}%
  \dateofbirth{2.1.1990}\placeofbirth{Dresden}%
  \course{Klinische Prognostik}\discipline{Individualprognose}%
  \and%
  Donald Duck\matriculationnumber{87654321}%
  \dateofbirth{1.2.1990}\placeofbirth{Berlin}%
  \course{Statistische Prognostik}\discipline{Makrosoziologische Prognosen}%
}\matriculationyear{2010}\issuedate{1.4.2015}\duedate{1.10.2015}
\supervisor{Dagobert Duck \and Mac Moneysac}\professor{Prof. Dr. Kater Karlo}
\referee{Prof. Dr. Kater Karlo}\chairman{Prof. Dr. Primus von Quack}
\taskform[pagestyle=empty]{%
  Momentan ist das besagte Thema in aller Munde. Insbesondere wird es
  gerade in vielen~-- wenn nicht sogar in allen~-- Medien diskutiert.
  Es ist momentan noch nicht abzusehen, ob und wann sich diese
  Situation ändert. Eine kurzfristige Verlagerung aus dem Fokus
  der Öffentlichkeit wird nicht erwartet.

  Als Ziel dieser Arbeit soll identifiziert werden, warum das Thema
  gerade so omnipräsent ist und wie dieser Effekt abgeschwächt werden
  könnte. Zusätzlich sind Methoden zu entwickeln, mit denen sich ein
  ähnlicher Vorgang zukünftig vermeiden lässt.
}{%
  \item Recherche \& Analyse
  \item Entwicklung eines Konzeptes \& Anwendung der entwickelten Methodik
  \item Dokumentation und grafische Aufbereitung der Ergebnisse
}
\end{document}
```



### **Aufgabenstellung für die Anfertigung einer Master-Arbeit**

Studiengang:	Klinische Prognostik	Statistische Prognostik
Studienrichtung:	Individualprognose	Makrosoziologische Prognosen
Name:	<b>Mickey Mouse</b>	<b>Donald Duck</b>
Matrikelnummer:	12345678	87654321
Immatrikulationsjahr:	2010	2010
Titel:	<b>Entwicklung eines optimalen Verfahrens zur Eroberung des Geldspeichers in Entenhausen</b>	

### **Ziele der Arbeit**

Momentan ist das besagte Thema in aller Munde. Insbesondere wird es gerade in vielen – wenn nicht sogar in allen – Medien diskutiert. Es ist momentan noch nicht abzusehen, ob und wann sich diese Situation ändert. Eine kurzfristige Verlagerung aus dem Fokus der Öffentlichkeit wird nicht erwartet.

Als Ziel dieser Arbeit soll identifiziert werden, warum das Thema gerade so omnipräsent ist und wie dieser Effekt abgeschwächt werden könnte. Zusätzlich sind Methoden zu entwickeln, mit denen sich ein ähnlicher Vorgang zukünftig vermeiden lässt.

### **Schwerpunkte der Arbeit**

- Recherche & Analyse
- Entwicklung eines Konzeptes & Anwendung der entwickelten Methodik
- Dokumentation und grafische Aufbereitung der Ergebnisse

Gutachter:	Prof. Dr. Kater Karlo
Betreuer:	Dagobert Duck Mac Moneysac
Ausgehändigt am:	1. April 2015
Einzureichen am:	1. Oktober 2015

Prof. Dr. Primus von Quack  
Prüfungsausschussvorsitzender

Prof. Dr. Kater Karlo  
Betreuender Hochschullehrer

## 6.5 Gutachten

Nach dem Laden des Paketes `tudscrsupervisor` kann ein Gutachten für eine wissenschaftliche Arbeit mit der `evaluation`-Umgebung oder dem Befehl `\evaluationform` erstellt werden.

```
\documentclass[ngerman]{tudscrartcl}
\iftutex
  \usepackage{fontspec}
\else
  \usepackage[T1]{fontenc}
  \usepackage[ngerman=ngerman-x-latest]{hyphsubst}
\fi
\usepackage{babel}
\usepackage{isodate}
\usepackage{tudscrsupervisor}
\usepackage{enumitem}\setlist{noitemsep}
\begin{document}
\faculty{Juristische Fakultät}\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}\chair{Lehrstuhl für Kriminalprognose}
\date{16.10.2015}
\author{Mickey Mouse}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}
\thesis{diploma}\graduation[Dipl.-Ing.]{Diplomingenieur}
\matriculationnumber{12345678}\matriculationyear{2010}
\dateofbirth{2.1.1990}\placeofbirth{Dresden}
\referee{Dagobert Duck}
\evaluationform[pagestyle=empty]{%
  Als Ziel dieser Arbeit sollte identifiziert werden, warum das Thema
  gerade so omnipräsent ist und wie sich dieser Effekt abschwächen
  ließe. Zusätzlich waren Methoden zu entwickeln, mit denen ein
  ähnlicher Vorgang zukünftig vermieden werden könnte.
}{%
  Die Arbeit gliedert sich in mehrere Kapitel auf unzähligen Seiten.
  In den ersten beiden Kapiteln wird dies und das besprochen. Im
  darauffolgenden auch noch jenes. Im vorletzten Kapitel wird alles
  betrachtet und im letzten erfolgt Zusammenfassung und Ausblick.
}{%
  Die dargestellten Ergebnisse der vorliegenden Arbeit genügen den in
  der Aufgabenstellung formulierten Arbeitsschwerpunkten. Sowohl
  Form, Ausdruck und Terminologie als auch Struktur befriedigen die
  Anforderungen an eine wissenschaftliche Arbeit. Die einzelnen
  Kapitel können hinsichtlich ihrer inhaltlichen Tiefe in Relation zu
  dem dazu benötigten Umfang als ausgewogen betrachtet werden. Die
  verwendeten Grafiken und Tabellen sind sehr gut aufbereitet und
  tragen insgesamt zum besseren Verständnis bei. Fachliche
  Begrifflichkeiten werden sehr sauber verwendet, orthografische
  Mängel sind keine zu finden.
}{1,0 (sehr gut)}
\end{document}
```



### **Gutachten für die Diplomarbeit**

Name: Mickey Mouse  
Matrikelnummer: 12345678  
Immatrikulationsjahr: 2010  
Titel: Entwicklung eines optimalen Verfahrens zur Eroberung des  
Geldspeichers in Entenhausen

### **Aufgabenstellung**

Als Ziel dieser Arbeit sollte identifiziert werden, warum das Thema gerade so omnipräsent ist und wie sich dieser Effekt abschwächen ließe. Zusätzlich waren Methoden zu entwickeln, mit denen ein ähnlicher Vorgang zukünftig vermieden werden könnte.

### **Inhalt**

Die Arbeit gliedert sich in mehrere Kapitel auf unzähligen Seiten. In den ersten beiden Kapiteln wird dies und das besprochen. Im darauffolgenden auch noch jenes. Im vorletzten Kapitel wird alles betrachtet und im letzten erfolgt Zusammenfassung und Ausblick.

### **Bewertung**

Die dargestellten Ergebnisse der vorliegenden Arbeit genügen den in der Aufgabenstellung formulierten Arbeitsschwerpunkten. Sowohl Form, Ausdruck und Terminologie als auch Struktur befriedigen die Anforderungen an eine wissenschaftliche Arbeit. Die einzelnen Kapitel können hinsichtlich ihrer inhaltlichen Tiefe in Relation zu dem dazu benötigten Umfang als ausgewogen betrachtet werden. Die verwendeten Grafiken und Tabellen sind sehr gut aufbereitet und tragen insgesamt zum besseren Verständnis bei. Fachliche Begrifflichkeiten werden sehr sauber verwendet, orthografische Mängel sind keine zu finden.

Die Arbeit wird mit der Note **1,0 (sehr gut)** bewertet.

Dresden, 16. Oktober 2015

Dagobert Duck  
Gutachter

## 6.6 Aushang

Das Paket `tudscrsupervisor` stellt die Umgebung `notice` für das Anfertigen allgemeiner Aushänge sowie den Befehl `\noticeform` für die Ausschreibung wissenschaftlicher Arbeiten bereit.

```
\documentclass[ngerman]{tudscrreprt}
\iftutex
  \usepackage{fontspec}
\else
  \usepackage[T1]{fontenc}
  \usepackage[ngerman=ngerman-x-latest]{hyphsubst}
\fi
\usepackage{babel}
\usepackage{isodate}
\usepackage{tudscrsupervisor}
\usepackage{enumitem}\setlist{noitemsep}
\begin{document}
\faculty{Juristische Fakultät}\department{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}\chair{Lehrstuhl für Kriminalprognose}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}\date{16.10.2015}
\contactperson{%
  Dagobert Duck\emailaddress{dagobert.duck@tu-dresden.de}
  \office{Dingens-Bau, Zimmer~08}\telephone{+49 351 463-12345}
\and%
  Mac Moneysac\emailaddress{mac.moneysac@tu-dresden.de}
  \office{Dingens-Bau, Zimmer~15}\telephone{+49 351 463-54321}
}
\noticeform[Angebot für eine Studien-/Diplomararbeit,pagestyle=empty]{%
  Momentan ist das besagte Thema in aller Munde. Insbesondere wird es
  gerade in vielen~-~ wenn nicht sogar in allen~-~ Medien diskutiert.
  Es ist momentan noch nicht abzusehen, ob und wann sich diese
  Situation ändert. Eine kurzfristige Verlagerung aus dem Fokus der
  Öffentlichkeit wird nicht erwartet.

  Als Ziel dieser Arbeit soll identifiziert werden, warum das Thema
  gerade so omnipräsent ist und wie dieser Effekt abgeschwächt werden
  könnte. Zusätzlich sind Methoden zu entwickeln, mit denen sich ein
  ähnlicher Vorgang zukünftig vermeiden lässt.
  \begin{center}
  \medskip\includegraphics[width=.7\linewidth]{DDC-21}
  \renewcommand*{\figureformat}{\figurename}
  \captionof{figure}{Thematisch passendes Bild}
  \end{center}
}{%
  \item Recherche \& Analyse
  \item Entwicklung eines Konzeptes \& Anwendung der entwickelten Methodik
  \item Dokumentation und grafische Aufbereitung der Ergebnisse
}
\end{document}
```

16. Oktober 2015

## Angebot für eine Studien-/Diplomarbeit

### Entwicklung eines optimalen Verfahrens zur Eroberung des Geldspeichers in Entenhausen

Momentan ist das besagte Thema in aller Munde. Insbesondere wird es gerade in vielen – wenn nicht sogar in allen – Medien diskutiert. Es ist momentan noch nicht abzusehen, ob und wann sich diese Situation ändert. Eine kurzfristige Verlagerung aus dem Fokus der Öffentlichkeit wird nicht erwartet.

Als Ziel dieser Arbeit soll identifiziert werden, warum das Thema gerade so omnipräsent ist und wie dieser Effekt abgeschwächt werden könnte. Zusätzlich sind Methoden zu entwickeln, mit denen sich ein ähnlicher Vorgang zukünftig vermeiden lässt.

**DRESDEN**  
concept

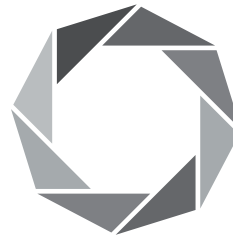


Abbildung: Thematisch passendes Bild

#### Schwerpunkte der Arbeit

- Recherche & Analyse
- Entwicklung eines Konzeptes & Anwendung der entwickelten Methodik
- Dokumentation und grafische Aufbereitung der Ergebnisse

#### Ansprechpartner

Dagobert Duck  
dagobert.duck@tu-dresden.de  
Dingens-Bau, Zimmer 08  
+49 351 463-12345

Mac Moneysac  
mac.moneysac@tu-dresden.de  
Dingens-Bau, Zimmer 15  
+49 351 463-54321

## 6.7 Poster

Mit der Klasse `tudscrposter` lässt sich ein Poster im Corporate Design der Technischen Universität Dresden erstellen. Dabei ist die Angabe des gewünschten Papierformates sowie der passenden Schriftgröße zu beachten.

```
\RequirePackage{fix-cm}
\documentclass[
  ngerman,%
  paper=A1,fontsize=22pt,%
  cdfoot=5ex,ddcfoot%
]{tudscrposter}
\iftutex
  \usepackage{fontspec}
\else
  \usepackage[T1]{fontenc}
  \usepackage[ngerman=ngerman-x-latest]{hyphsubst}
\fi
\usepackage{babel}
\usepackage{blindtext}
\usepackage{multicol}
\begin{document}
\faculty{Juristische Fakultät}\department[ ]{Fachrichtung Strafrecht}
\institute{Institut für Kriminologie}\chair[ ]{Lehrstuhl für Kriminalprognose}
\title{%
  Entwicklung eines optimalen Verfahrens zur Eroberung des
  Geldspeichers in Entenhausen
}\date{18.02.2015}
\contactperson{%
  Mickey Mouse\emailaddress{mickey.mouse@tu-dresden.de}%
  \office{Büro 58}\telephone{463--xxxxx}%
}
\professor{Prof. Dr. Kater Karlo}
\maketitle
\begin{abstract}%[columns=2]
\noindent\blindtext
\end{abstract}

\begin{multicols}{2}[\section*{Test}]
\blindtext[3]
\end{multicols}
\bigskip

\begin{center}
\includegraphics[width=.7\linewidth]{TUD-black}
\end{center}

\begin{multicols}{2}[\section*{Test}]
\blindmathtrue
\blindtext[1]
\end{multicols}
\end{document}
```



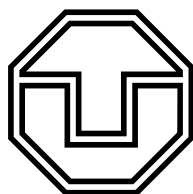
## Entwicklung eines optimalen Verfahrens zur Eroberung des Geldspeichers in Entenhausen

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburnn? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

### Test

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburnn? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburnn? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie

monisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburnn? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.



### Test

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld.  $\sin^2(\alpha) + \cos^2(\beta) = 1$ . Der Text gibt lediglich den Grauwert der Schrift an  $E = mc^2$ . Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburnn? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen.  $\sqrt{a} \cdot \sqrt{b} = \sqrt{ab}$ . An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die

Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft.  $\frac{\sqrt{a}}{\sqrt{b}} = \sqrt{\frac{a}{b}}$ . Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein.  $a\sqrt{b} = \sqrt{a}b$ . Er muss keinen Sinn ergeben, sollte aber lesbar sein.  $d\Omega = \sin \theta d\theta d\varphi$ . Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.

### Kontakt

Technische Universität Dresden  
Juristische Fakultät  
Institut für Kriminologie  
Prof. Dr. Kater Karlo

### Ansprechpartner

Mickey Mouse  
micky.mouse@tu-dresden.de  
Büro 58  
463-xxxxx





## Kapitel 7

# Tutorials

In diesem Kapitel werden weiterführende Anwendungsbeispiele bereitgestellt. Diese Tutorials sind nicht unmittelbar im Handbuch enthalten sondern werden als externe Dateien bereitgehalten, welche direkt via Hyperlink geöffnet werden können.

## Leitfaden für eine wissenschaftlichen Arbeit

Die meisten Anwender der TUD-Script-Klassen sind Studenten oder angehörige der Technischen Universität Dresden, die ihre ersten Schritte mit  $\LaTeX 2_{\epsilon}$  beim Verfassen einer wissenschaftlichen Arbeit oder ähnlichem machen. Während der Einstiegsphase in  $\LaTeX 2_{\epsilon}$  kann ein Anfänger sehr schnell aufgrund der großen Anzahl an empfohlenen Pakete sowie der teilweise diametral zueinander stehenden Hinweise überfordert sein. Mit dem Tutorial **treatise.pdf** soll versucht werden, ein wenig Licht ins Dunkel zu bringen. Es erhebt jedoch keinerlei Anspruch, vollständig oder perfekt zu sein. Einige der darin vorgestellten Möglichkeiten lassen sich mit Sicherheit auch anders, einfacher und/oder besser lösen. Dennoch ist es gerade für Neulinge – vielleicht auch für den einen oder anderen  $\LaTeX$ -Veteran – als Leitfaden für die Erstellung einer wissenschaftlichen Arbeit gedacht.

## Ein Beitrag zum mathematischen Satz in $\LaTeX 2_{\epsilon}$

Das Tutorial **mathtype.pdf** richtet sich an alle Anwender, die in ihrem  $\LaTeX$ -Dokument mathematische Formeln setzen wollen. In diesem wird ausführlich darauf eingegangen, wie mit wenigen Handgriffen ein typografisch sauberer Mathematiksatz zu bewerkstelligen ist.

## Änderung der Trennzeichen im Mathematikmodus

Sollen beim Verfassen eines  $\LaTeX$ -Dokumentes Daten in einem Zahlenformat importiert werden, welches nicht den Gepflogenheiten der Dokumentsprache entspricht, kommt es meist zu unschönen Ergebnissen bei der Ausgabe. Einfachstes Beispiel sind Daten, in denen als Dezimaltrennzeichen ein Punkt verwendet wird, wie es im englischsprachigen Raum der Fall ist. Sollen diese in einem Dokument deutscher Sprache eingebunden werden, müssten diese normalerweise allesamt angepasst und das ursprüngliche Dezimaltrennzeichen durch ein Komma ersetzt werden. Dieser Schritt wird mit dem TUD-Script-Paket **mathswap** automatisiert. Wie dies genau funktioniert, wird im Tutorial **mathswap.pdf** erläutert.

## Kapitel 8

# Benötigte, unterstützte und empfehlenswerte Pakete

## 8.1 Notwendige und ergänzende Pakete für TUD-Script

Diese Pakete werden von den TUD-Script-Klassen zwingend benötigt und eingebunden. Möchten Sie eines der im Folgenden aufgezählten Pakete selber mit bestimmten Optionen nutzen, so sollten diese bereits *vor* dem Laden der Dokumentklasse an das Paket weitergereicht werden, falls bei der folgenden Beschreibung des Paketes nichts anderweitiges angegeben wird.

**Beispiel:** Das Weiterreichen von Optionen an Pakete muss folgendermaßen erfolgen:

```
\PassOptionsToPackage{<Paketoptionen>}{<Paket>}
\documentclass[<Klassenoptionen>]{tudscr. . .}
```

v2.02

### koma-script, typearea, scrbase, sclayer-scrpage, scrextend

Das KOMA-Script-Bundle ist die zentrale Grundlage für TUD-Script. Neben den Klassen **scrbook**, **scrreprt** und **scrartcl** wird das Paket **scrbase** benötigt. Dieses erlaubt das Definieren von Klassenoptionen im Stil von KOMA-Script, welche auch noch nach dem Laden der Klasse mit den Befehlen `\TUDoption` und `\TUDoptions` geändert werden können. Für die Bereitstellung der `tudheadings`-Seitenstile ist das Paket **sclayer-scrpage** notwendig. Wenn es nicht durch den Anwender – mit beliebigen Optionen – geladen wird, erfolgt dies am Ende der Präambel automatisch durch TUD-Script.

v2.06

### opensans, iwona

Das Paket **opensans** stellt die Schriftfamilie **Open Sans** sowohl für den Fließtext als auch den mathematischen Satz zur Verfügung. Es enthält alle nötigen Schriftschnitte sowohl im Type1- als auch im OpenType-Format. Da die Schriftfamilie in der aktuellen Version keine mathematischen Glyphen bereitstellt, werden die Pakete **mathastext** und **iwona** zusätzlich genutzt, um zumindest einen halbwegs erträglichen mathematischen Satz mit **Open Sans** zu ermöglichen. Werden dabei zusätzliche Symbole benötigt, wird empfohlen, auf das Paket **amssymb** zu verzichten und anstelle dessen **mdsymbol** zu laden.

v2.06

### mweights

In  $\text{\TeX}2_{\epsilon}$  existieren die Schriftfamilien für Serifenschriften (`\rmfamily`), serifenlose Schriften (`\sffamily`) sowie die Schreibmaschinenschriften (`\ttfamily`). Deren Schriftstärke wird für gewöhnlich mit den beiden Befehlen `\mddefault` und `\bfdefault` einheitlich festgelegt. Bei der Verwendung unterschiedlicher Schriftpakete kann es unter Umständen zu Problemen bei den Schriftstärken kommen. Diese Paket erlaubt die individuelle Definition der Schriftstärke für jede der drei Schriftfamilien.

### geometry

Das Paket wird zum Festlegen der Seitenränder respektive des Satzspiegels verwendet. *Ein Weiterreichen zusätzlicher Optionen an das Paket wird dringlich nicht empfohlen.*

Achtung!

### graphicx

Dies ist das De-facto-Standard-Paket zum Einbinden von Grafiken. Zum Setzen des Logos der Technischen Universität Dresden im Kopf sowie aller weitere Abbildungen und Logos wird `\includegraphics` genutzt.

### xcolor

Damit werden die Farben des Corporate Designs zur Verwendung im Dokument definiert. Genaueres ist bei der Beschreibung von **tudscrcolor** in Unterkapitel 5.1 zu finden.

### kvsetkeys

Das von **scrbase** geladene Paket **keyval** ermöglicht das Definieren von Klassen- und Paketoptionen sowie Parametern nach dem Schlüssel-Wert-Prinzip. Mit diesem Paket kann das Verhalten für unbekannte Schlüssel festgelegt werden.

### etoolbox, xpatch, letltxmacro, etexcmds

Diese Pakete stellen viele Funktionen zum Testen und zur Ablaufkontrolle bereit. Weiterhin wird das Manipulieren vorhandener Makros ermöglicht.

### environ

Es wird eine verbesserte Deklaration von Umgebungen ermöglicht, bei der auch beim Abschluss der Umgebung auf die übergebenen Parameter zugegriffen werden kann.

### trimspaces

Bei mehreren Eingabefeldern für den Anwender werden die Argumente mithilfe dieses Paketes um eventuell angegebene, unnötige Leerzeichen befreit.

## Durch TUD-Script direkt unterstützte Pakete

Einige der unter Unterkapitel 8.2 beschriebenen Pakete werden durch TUD-Script direkt unterstützt und erweitern dessen Funktionalität. Dies sind namentlich **hyperref**, **multicol**, **quoting**, **ragged2e** und **crop** sowie **isodate** respektive **datetime2**. Weitere Informationen dazu ist den nachfolgenden Beschreibung des jeweiligen Paketes zu entnehmen.

## 8.2 Empfehlenswerte Pakete

In diesem Unterkapitel wird eine Vielzahl an Paketen – zumeist kurz – vorgestellt, welche sich für mich persönlich bei der Arbeit mit  $\LaTeX$  bewährt haben. Einige davon werden außerdem im Tutorial **treatise.pdf** in ihrer Anwendung beschrieben. Für detaillierte Informationen sowie bei Fragen zu den einzelnen Paketen sollte die jeweilige Dokumentation<sup>47</sup> zu Rate gezogen werden, das Lesen der hier gegebenen Kurzbeschreibung ersetzt dies in keinem Fall.

### 8.2.1 Pakete zur Verwendung in jedem Dokument

Die hier vorgestellten Pakete gehören meiner Meinung nach in die Präambel eines jeden Dokumentes. Die Dokumentsprache sollte in jedem Fall mit **babel** oder **polyglossia** definiert werden – auch wenn dies Englisch ist. Für deutschsprachige Dokumente ist für eine annehmbare Worttrennung beim Einsatz von  $\pdf\LaTeX$  das Paket **hyphsubst** unbedingt zu verwenden.

v2.02

#### fontenc, fontspec

Die Zeichensatzkodierung des Ausgabefonts sollte immer festgelegt werden. Für  $\pdf\LaTeX$  ist die Ausgabe als 7-bit-kodierte Schrift in der Voreinstellung gewählt, was unter anderem dazu führt, dass keine echten Umlaute im erzeugten PDF-Dokument verwendet werden. Um auf 8-bit-Schriften zu schalten, ist `\usepackage[T1]{fontenc}` zu nutzen.

Für die Unicode-Textsatzsysteme  $\text{Lua}\LaTeX$  oder  $\text{X}\LaTeX$  sollte stattdessen das Paket **fontspec** verwendet werden. Damit können Systemschriften im OpenType-Format und einer beliebigen Zeichensatzkodierung eingebunden werden, womit sich die Auswahl der verwendbaren Schriften stark erweitert. Das Paket wird durch TUD-Script unterstützt.

#### microtype

Dieses Paket kümmert sich um den optischen Randausgleich<sup>48</sup> und das Nivellieren der Wortzwischenräume<sup>49</sup> im Dokument. Es funktioniert nicht mit der klassischen  $\text{T}\LaTeX$ -Engine, wohl jedoch mit  $\text{pdf}\LaTeX$  oder  $\text{Lua}\LaTeX$  sowie  $\text{X}\LaTeX$ .

<sup>47</sup>Kommandozeile/Terminal: `texdoc <Paketname>`

<sup>48</sup>englisch: protrusion, margin kerning

<sup>49</sup>englisch: font expansion

## **babel, polyglossia**

Mit dem Paket **babel** erfolgt die Einstellung der im Dokument verwendeten Sprache(n). Bei mehreren angegebenen Sprachen ist die zuletzt geladene die Hauptsprache des Dokumentes. Die gewünschten Sprachen sollten nicht als Paketoption sondern als Klassenoption und gesetzt werden, damit auch andere Pakete auf die Spracheinstellungen zugreifen können. Für deutschsprachige Dokumente ist die Option `ngerman` für die neue oder `german` für die alte deutsche Rechtschreibung zu verwenden.

Mit dem Laden von **babel** und der dazugehörigen Sprachen werden sowohl die Trennmuster als auch die sprachabhängigen Bezeichner angepasst. Von einer Verwendung der obsoleten Pakete **german** beziehungsweise **ngerman** anstelle von **babel** wird abgeraten. Für **Lua $\LaTeX$**  und **X $\LaTeX$**  kann das Paket **polyglossia** genutzt werden.

## **hyphsubst, dehyph-exptl**

Die möglichen Trennstellen von Wörtern wird von  $\LaTeX$  mithilfe eines Algorithmus berechnet. Dieser wird für deutschsprachige Texte mit dem Paket **hyphsubst** entscheidend verbessert. **Lua $\LaTeX$**  und **X $\LaTeX$**  nutzen diese Trennmuster automatisch, für **pdf $\LaTeX$**  müssen diese mit folgendermaßen eingebunden werden:

```
\usepackage[ngerman=ngerman-x-latest]{hyphsubst}
```

In Unterkapitel 9.3 wird genauer auf das Zusammenspiel von **hyphsubst** und **babel** sowie **fontenc** eingegangen, ein Blick dahin wird dringend empfohlen. Zusätzliche werden dort weitere Hinweise für eine verbesserte Worttrennung gegeben.

## **8.2.2 Pakete zur situativen Verwendung**

Die nachfolgenden Pakete sollten nicht zwangsweise in jedem Dokument geladen werden sondern nur, falls dies auch tatsächlich notwendig ist. Zur besseren Übersicht wurde versucht, diese thematisch passend zu gruppieren. Daraus lässt sich keinerlei Wertung bezüglich ihrer Nützlichkeit oder meiner persönlichen Wertschätzung ableiten.

### **Typografie und Layout**

Neben dem zuvor beschriebenen Paket **microtype**, welches verantwortlich für mikrotypografische Feinheiten ist, existieren weitere Pakete, die vorrangig die Makrotypografie adressieren.

#### **setspace**

Die Vergrößerung des Zeilenabstandes wird:

- I. viel zu häufig und völlig unnötig gefordert und
- II. schließlich auch noch zu groß gewählt.

Die Forderung nach Erhöhung des Zeilenabstandes – in der Typografie als Durchschuss bezeichnet – kommt aus den Zeiten der Textverarbeitung mit der Schreibmaschine. Ein einzeiliger Zeilenabstand bedeutete hier, dass die Unterlängen der oberen Zeile genau auf der Höhe der Oberlängen der folgenden Zeile lagen. Ein anderthalbzeiliger Zeilenabstand erzielte hier somit einen akzeptablen Durchschuss. Eine Erhöhung des Durchschusses bei der Verwendung von  $\LaTeX$  ist an und für sich nicht notwendig. Sinnvoll ist dies nur, wenn im Fließtext serifenlose Schriften zum Einsatz kommen, um die damit verbundene schlechte Lesbarkeit etwas zu verbessern.

Ist die Erhöhung des Durchschusses wirklich notwendig, sollte das Paket **setspace** genutzt werden. Dieses stellt den Befehl `\setstretch{<Faktor>}` zur Verfügung, mit

dem der Durchschuss respektive Zeilenabstand angepasst werden kann. Der Wert des Faktors ist standardmäßig auf 1 gestellt und sollte maximal bis 1.25 vergrößert werden. Der Befehl `\onehalfspacing` aus diesem Paket setzt diesen Wert auf eben genau 1.25. Allerdings ist hier anzumerken, dass die Vergrößerung des Zeilenabstandes – so wie ich es mir angelesen habe – aus der Sicht eines Typographen keine Spielerei ist sondern vielmehr allein der Lesbarkeit des Textes dient und möglichst gering ausfallen sollte.

Ziel ist es, beim Lesen nach dem Beenden einer Zeile das Auffinden der neuen Zeile zu vereinfachen. Bei Serifen ist dies durch die Betonung der Grundlinie sehr gut möglich. Bei serifenlosen Schriften – wie der im Corporate Design der Technischen Universität Dresden verwendeten **Open Sans** – ist dies schwieriger und ein erweiterter Abstand der Zeilen kann hierbei hilfreich sein. Jedoch sollte nicht nach dem Motto „viel hilft viel“ verfahren werden. Für dieses Dokument wurde `\setstretch{1.1}` für den Zeilenabstand gewählt. Weitere Tipps sind in Unterkapitel 9.7 und Unterkapitel 9.8 zu finden.

### multicol

Hiermit kann jeglicher beliebiger Inhalt in zwei oder mehr Spalten ausgegeben werden, wobei – im Gegensatz zum normalen zweispaltigen Satz über die KOMA-Script-Option `twocolumn` – für einen Spaltenausgleich gesorgt wird. Unterstützt wird das Paket innerhalb der Umgebungen `abstract` und `tudpage`.

### balance

Dieses Paket ermöglicht einen Spaltenausgleich im zweispaltigen Satz auf der letzten Dokumentseite. Alternativ dazu kann auch **multicol** verwendet werden.

### isodate, datetime2

Mit `\printdate{<Datum>}` formatiert das Paket **isodate** die Ausgabe eines Datums automatisch in ein spezifiziertes Format. Die Datumsangabe kann dabei im deutschen, englischen oder ISO-Format erfolgen. Alternativ kann auch das Paket **datetime2** genutzt werden, welches zwingend die Eingabe im ISO-Format erfordert. Wird eines der beiden Pakete geladen, werden alle Datumsfelder, welche durch die TUD-Script-Klassen definiert wurden,<sup>50</sup> im durch das jeweilige Paket definierten Ausgabeformat ausgegeben.

### quoting

$\LaTeX$  bietet von Haus aus *zwei* verschiedene Umgebungen – `quote` und `quotation` – für Zitate und ähnliches an. Allerdings werden durch beide Umgebungen die KOMA-Script-Option `parskip=<Methode>` ignoriert. Mit der Umgebung `quoting` aus dem gleichnamigen Paket lässt sich dieses Problem umgehen. Wird das Paket geladen, wird diese innerhalb der `abstract`-Umgebung verwendet.

### csquotes

Das Paket stellt unter anderem den Befehl `\enquote{<Zitat>}` zur Verfügung, welcher Anführungszeichen in Abhängigkeit der gewählten Sprache setzt. Zusätzlich werden weitere Kommandos und Optionen für die spezifischen Anforderungen des Zitierens bei wissenschaftlichen Arbeiten angeboten. Außerdem wird es durch **biblatex** unterstützt und sollte zumindest bei dessen Verwendung geladen werden.

v2.02

### noindentafter

Mit diesem Paket lassen sich automatische Absatzeinzüge für selbst zu bestimmende Befehle und Umgebungen unterdrücken.

### ragged2e

Das Paket verbessert den Flattersatz, indem für diesen die Worttrennung aktiviert wird.

---

<sup>50</sup>`\date`, `\dateofbirth` und `\defensedate` sowie aus **tudscrsupervisor** `\duedate` und `\issuedate`

**fnpct**

Diese Paket sorgt zum einen für das Einhalten der richtigen Reihenfolge von Satzzeichen und Fußnoten und zum anderen wird das typografisch korrekte Setzen mehrerer, nacheinander folgender Fußnoten unterstützt.

**xspace, xpunctuate**

Mit **xspace** kann bei der Definition eigener Makros der Befehl `\xspace` genutzt werden. Dieser setzt ein gegebenenfalls notwendiges Leerzeichen automatisch. In Unterkapitel 9.13 ist die Definition eines solchen Befehls exemplarisch ausgeführt. Durch das Paket **xpunctuate** wird **xspace** um die Beachtung von Interpunktionen erweitert.

**ellipsis**

In  $\LaTeX$  folgen den Befehlen für Auslassungspunkte (`\dots` und `\textellipsis`) *immer* ein Leerzeichen. Dies kann unter Umständen unerwünscht sein. Mit dem Paket **ellipsis** wird das nachfolgende Leerzeichen – im Gegensatz zum Standardverhalten – nur gesetzt, wenn ein Satzzeichen und kein Buchstabe folgt, *siehe auch* Unterkapitel 9.15.

**DeLig, selnolig**

Hierbei handelt es sich um ein Java-Script, das anhand eines Wörterbuches falsche Ligaturen innerhalb eines Dokumentes automatisiert entfernt. Diese müssten – insbesondere in deutschen Texten aufgrund der vielen Komposita – für einen guten Satz manuell aufgelöst werden.<sup>51</sup> Mit **Lua $\LaTeX$**  als Textsatzsystem kann auch **selnolig** alternativ dazu verwendet werden.

**Rechtschreibung**

Für die Rechtschreibkontrolle zeichnet im Normalfall der verwendete Editor verantwortlich. Dennoch gibt es einige wenige Pakete, welche sich diesem Thema widmen. Diese sind jedoch ausschließlich nutzbar, wenn als Textsatzsystem **Lua $\LaTeX$**  genutzt wird.

**lua-check-hyphen**

Hiermit lassen sich mit **Lua $\LaTeX$**  Trennstellen am Zeilenende zur Prüfung markieren. Zum Thema der *korrekten Worttrennung* sei außerdem auf Unterkapitel 9.3 verwiesen.

**spelling**

Wird **Lua $\LaTeX$**  als Textsatzsystem verwendet, wird mit diesem Paket der reine Textanteil aus dem  $\LaTeX$ -Dokument extrahiert – wobei Makros und aktive Zeichen entfernt werden – und in eine separate Textdatei geschrieben. Anschließend kann diese Datei mit einer externen Software zur Rechtschreibprüfung wie **GNU Aspell**, **Hunspell** oder **LanguageTool** analysiert und falsch geschriebene Wörter im PDF-Dokument hervorgehoben werden.

**Schriften und Sonderzeichen****Imodern**

Soll mit den klassischen  $\LaTeX$ -Standardschriften gearbeitet werden, empfiehlt sich die Verwendung des Paketes **Imodern**. Dieses verbessert die Darstellung der Computer Modern sowohl am Bildschirm als auch beim finalen Druck.

**cfr-lm**

Dieses experimentelle Paket liefert weitere Schriftschnitte für das Paket **Imodern**.

<sup>51</sup> Das sind „ff“, „fi“, „fl“, „ffi“, und „ffl“ bei den meisten  $\LaTeX$ -Schriften.



### **newtx, newtxmath**

Es werden einige alternative Schriften sowohl für den Fließtext (*Times* und *Helvetica*) als auch den Mathematikmodus bereitgestellt.

### **libertine**

Das Paket stellt die Schriften Linux Libertine und Linux Biolinum zur Verfügung. Um diese Schriftart auch für den Mathematikmodus verwenden zu können, sollte **newtxmath** aus dem **newtx**-Bundle mit `\usepackage[libertine]{newtxmath}` in der Präambel eingebunden werden. Das Paket **libgreek** enthält griechische Buchstaben für Linux Libertine.

### **relsize**

Die Größe einer Textauszeichnung kann relativ zur aktuellen Schriftgröße gesetzt werden.

### **textcomp**

Es werden zusätzliche Symbole und Sonderzeichen wie beispielsweise das Promille- oder Eurozeichen sowie Pfeile für den Fließtext zur Verfügung gestellt.

Auch für (serifenlose) Mathematikschriften gibt es einige nützliche Pakete. Werden die Schriften des Corporate Designs genutzt, sei auf die Option `cdmath` verwiesen.

### **sansmathfonts, sansmath**

Sollten die normalen  $\text{\LaTeX}$ -Schriften Computer Modern verwendet werden, lässt sich dieses Paket zum serifenlosen Setzen mathematischer Ausdrücke nutzen. Ein alternatives Paket mit der gleichen Zielstellung ist **sansmath**

### **sfmath**

Dieses Paket verfolgt ein ähnliches Ziel, kann jedoch im Gegensatz zu **sansmath** nicht nur für Computer Modern sondern mit der entsprechenden Option auch für Latin Modern, Helvetica und Computer Modern Bright verwendet werden.

### **mathastext**

Mit dem Paket wird das Ziel verfolgt, aus der genutzten Schrift für den Fließtext alle notwendigen Zeichen für den Mathematiksatz zu extrahieren.

## **Mathematiksatz**

Dies sind Pakete, die Umgebungen und Befehle für den Mathematiksatz sowie das Setzen von Einheiten und Zahlen im Allgemeinen anbieten.

### **mathtools, amsmath**

Dieses Paket stellt für das De-facto-Standard-Paket **amsmath** für Mathematikumgebungen Bugfixes zur Verfügung und erweitert dieses.

### **bm**

Das Paket bietet mit `\bm` eine Alternative zu `\boldsymbol` im Mathematiksatz<sup>52</sup>. Sollten bei der Nutzung des Paketes Fehler auftreten, sei auf dessen Dokumentation und insbesondere auf das Makro `\bmmax` verwiesen.

Die korrekte Formatierung von Zahlen ist häufig ein Problem bei der Verwendung von  $\text{\LaTeX}$ . Insbesondere, wenn in einem deutschsprachigen Dokument Daten im englischsprachigen Format verwendet werden, kommt es zu Problemen. Dafür wird im TUD-Script-Bundle das Paket **mathswap** bereitgestellt. Dennoch gibt es zu diesem auch Alternativen.

### **ionumbers**

Dieses Paket ist mir tatsächlich erst bei der Arbeit an **mathswap** bekannt geworden. Es bietet mehr Funktionalitäten und kann als Alternative dazu betrachtet werden.

---

<sup>52</sup><http://tex.stackexchange.com/questions/3238>

## icomma

Wird im Mathematikmodus nach dem Komma ein Leerzeichen gesetzt, wird dies bei der Ausgabe beachtet. Der Verfasser muss sich demzufolge jederzeit selbst um die typografisch korrekte Ausgabe kümmern.

## ziffer

Für deutschsprachige Dokumente wird das Komma als Dezimaltrennzeichen zwischen zwei Ziffern definiert. Folgt dem Komma keine Ziffer, wird jederzeit der obligatorische Freiraum gesetzt, was meiner Meinung nach besser als das Verhalten von **icomma** ist.

Für das typografisch korrekte Setzen von Einheiten – ein halbes Leerzeichen zwischen Zahl und *aufrecht* gesetzter Einheit – gibt es zwei gut nutzbare Pakete.

## units

Dies ist ein einfaches und sehr zweckdienliches Paket zum Setzen von Einheiten und für die meisten Anforderungen völlig ausreichend.

## siunitx

Dieses Paket ist in seinem Umfang im Vergleich deutlich erweitert. Neben Einheiten können zusätzlich auch Zahlen typografisch korrekt gesetzt werden. Die Ausgabe lässt sich in vielerlei Hinsicht an individuelle Bedürfnisse anpassen. Für deutschsprachige Dokumenten sollte die Lokalisierung angegeben werden. Mehr dazu in Unterkapitel 9.16.

Weitere Hinweise zur mathematischen Typografie werden in Kapitel 7 gegeben.

## Listen

### enumitem

Das Paket **enumitem** erweitert die rudimentären Funktionalitäten der  $\LaTeX$ -Standardlisten `itemize`, `enumerate` sowie `description` und ermöglicht die individuelle Anpassung dieser durch die Bereitstellung vieler optionaler Parameter nach dem Schlüssel-Wert-Prinzip. Eine von mir sehr häufig genutzte Funktion ist beispielsweise die Entfernung des zusätzlichen Abstand zwischen den einzelnen Einträgen einer Liste mit `\setlist{noitemsep}`.

## Verzeichnisse aller Art

Neben dem Erstellen des eigentlichen Dokumentes sind für eine wissenschaftliche Arbeit meist auch allerhand Verzeichnisse gefordert. Fester Bestandteil ist dabei das Literaturverzeichnis, auch ein Abkürzungs- und Formelzeichen- beziehungsweise Symbolverzeichnis werden häufig gefordert. Gegebenenfalls wird auch noch ein Glossar benötigt. Hier werden die passenden Pakete vorgestellt. Sollen im Dokument komplette Quelltexte oder auch nur Auszüge daraus erscheinen und für diese auch gleich ein entsprechendes Verzeichnis generiert werden, so sei auf das Paket **listings** in Unterabschnitt 8.2.2 auf Seite 113 verwiesen.

### biblatex

Das Paket kann als legitimer Nachfolger zu  $\text{BibTeX}$  gesehen werden. Ähnlich dazu bietet **biblatex** die Möglichkeit, Literaturdatenbanken einzubinden und verschiedene Stile der Referenzierung und Darstellung des Literaturverzeichnisses auszuwählen.

Mit **biblatex** ist die Anpassung eines bestimmten Stiles wesentlich besser umsetzbar als mit  $\text{BibTeX}$ . Wird **biber** für die Sortierung des Literaturverzeichnisses genutzt, ist die Verwendung einer UTF-8-kodierten Literaturdatenbank problemlos möglich. In Verbindung mit **biblatex** wird die zusätzliche Nutzung des Paketes **csquotes** sehr empfohlen.

## glossaries, nomencl

Dies ist ein sehr mächtiges Paket zum Erstellen eines Glossars sowie Abkürzungs- und Symbolverzeichnis. Die mannigfaltige Anzahl an Optionen ist zu Beginn eventuell etwas abschreckend. Insbesondere wenn Verzeichnisse für Abkürzungen *und* Formelzeichen respektive Symbole notwendig sind, sollte dieses Paket in Erwägung gezogen werden.

Alternativ dazu kann für ein Symbolverzeichnis auch lediglich eine manuell gesetzte Tabelle genutzt werden. Das hierfür sehr häufig empfohlene Paket **nomencl** bietet meiner Meinung nach demgegenüber keinerlei Vorteile.

## acro, acronym

Soll lediglich ein Abkürzungsverzeichnis erstellt werden, ist dieses Paket die erste Wahl. Es stellt Befehle zur Definition von Abkürzungen sowie zu deren Verwendung im Text und zur sortierten Ausgabe eines Verzeichnisses bereit. Alternativ dazu kann das Paket **acronym** verwendet werden. Die Sortierung des Abkürzungsverzeichnisses muss hier allerdings manuell durch den Anwender erfolgen.

## Gleitobjekte

Es werden Pakete für die Beeinflussung von Aussehen, Beschriftung und Positionierung von Gleitobjekten vorgestellt. Unter Unterkapitel 9.23 sind außerdem Hinweise zur manuellen Manipulation der Gleitobjektplatzierung zu finden.

### caption

Mit der Option `captions=<Einstellung>` bieten die KOMA-Script-Klassen bereits einige Möglichkeiten zum Formatieren der Beschriftungen für Gleitobjekte. Dieses Paket ist daher meist nur in gewissen Ausnahmefällen für spezielle Anweisungen notwendig, allerdings auch bei der Verwendung unbedenklich.

### subcaption

Diese Paket kann zum einfachen Setzen von Unterabbildungen oder -tabellen mit den entsprechenden Beschriftungen genutzt werden. Das dazu alternative Paket **subfig** sollte vermieden werden, da es nicht mehr gepflegt wird und es mit diesem im Zusammenspiel mit anderen Paketen des Öfteren zu Problemen kommt. Sollte der Funktionsumfang von **subcaption** nicht ausreichen, kann anstelle dessen das Paket **floatrow** verwendet werden, welches ähnliche Funktionalitäten wie **subfig** bereitstellt.

### floatrow

Mit diesem Paket können global wirksame Einstellungen und Formatierungen für *alle* Gleitobjekte eines Dokumentes über die Paketoptionen oder mit `\floatsetup{. . .}` vorgenommen werden. So lässt sich unter anderem die verwendete Schrift innerhalb der Umgebungen `figure` und `table` mit `\floatsetup{font=. . .}` einstellen. Mit

```
\floatsetup[figure]{capposition=bottom}
\floatsetup[table]{capposition=top}
```

lässt sich automatisch das typografisch korrekte Setzen von Abbildungs*unterschriften* sowie Tabellen*überschriften* erzwingen – unabhängig von der Position des Befehls zur Beschriftung `\caption` innerhalb der jeweiligen Gleitobjektumgebung. Wird das Verhalten wie empfohlen eingestellt, sollte für eine gute vertikale Platzierung der Tabellenüberschriften zusätzlich die KOMA-Script-Option `captions=tableheading` genutzt werden.

### placeins

Mit diesem Paket kann die Ausgabe von Gleitobjekten vor Kapiteln und wahlweise Unterkapiteln erzwungen werden.

## flafter

Dieses Paket erlaubt die frühestmögliche Platzierung von Gleitobjekten im ausgegebenen Dokument erst an der Stelle ihres Auftretens im Quelltext. Diese werden dementsprechend nie vor ihrer Definition am Anfang der Seite erscheinen.

## Tabellen

Für den Tabellensatz werden standardmäßig die Umgebungen `tabbing` und `tabular` respektive `tabular*` bereitgestellt, welche in ihrer Funktionalität für einen qualitativ hochwertigen Tabellensatz meist nicht ausreichen. Deshalb werden hier zusätzliche Pakete vorgestellt.

### array

Dieses Paket ermöglicht mit `\newcolumn` das Erstellen neuer Spaltentypen und die erweiterte Definition von Tabellenspalten (`>{...}<Spaltentyp><{...}<`), wobei mithilfe sogenannter „Hooks“ vor und nach Einträgen innerhalb einer Spalte gezielt Anweisungen gesetzt werden können. Die Zeilenhöhe lässt sich mit `\extrarowheight` ändern.

### booktabs

Für einen guten Tabellensatz mit  $\LaTeX$  gibt es bereits zahlreiche Tipps<sup>53</sup> im Internet zu finden. Zwei Regeln sollten dabei definitiv beachtet werden:

- I. keine vertikalen Linien
- II. keine doppelten Linien

Das Paket **booktabs** (deutsche Dokumentation **booktabs-de**) ist für den Satz von hochwertigen Tabellen eine große Hilfe und stellt die Befehle `\toprule`, `\midrule` sowie `\cmidrule` und `\bottomrule` für unterschiedliche horizontale Linien bereit.

### widetable

Mit der Standard- $\LaTeX$ -Umgebung `tabular*` kann eine Tabelle mit einer definierten Breite gesetzt werden. Dieses Paket stellt die Umgebung `widetable` zur Verfügung, die als Alternative genutzt werden kann und eine symmetrische Tabelle erzeugt.

### tabularx

Auch mit diesem Paket kann die Gesamtbreite einer Tabelle spezifiziert werden. Dafür wird der Spaltentyp `X` definiert, welcher als Argument der `tabularx`-Umgebung beliebig häufig angegeben werden kann (`\begin{tabularx}{<Breite>}{<Spalten>}`). Die `X`-Spalten ähneln denen vom Typ `p<Breite>`, wobei die Breite dieser aus der gewünschten Tabellengesamtbreite abzüglich dem benötigten Platz der gegebenenfalls vorhandenen Standardspalten automatisch berechnet wird.

### tabulary

Dies ist ein weiteres Paket zur automatischen Berechnung von Spaltenbreiten. Der zur Verfügung stehende Platz – gewünschte Gesamtbreite abzüglich der notwendigen Breite für Standardspalten – wird jedoch nicht wie bei der `tabularx` auf alle Spalten gleichmäßig verteilt sondern in der `tabulary`-Umgebung für die Spaltentypen `LCRJ` anhand ihres Zellinhaltes gewichtet vergeben. (`\begin{tabulary}{<Breite>}{<Spalten>}`).

### longtable, ltxtable, ltablex

Sollen mehrseitige Tabellen mit Seitenumbruch erstellt werden, ist dieses Paket das Mittel der ersten Wahl. Für die Kombination mehrseitiger Tabellen mit einer `tabularx`-Umgebung können die Pakete **ltablex** oder besser noch **ltxtable** verwendet werden. Alternativ dazu lässt sich auch **tabu** nutzen.

<sup>53</sup><http://userpage.fu-berlin.de/latex/Materialien/tabsatz.pdf>

## multirow

Es wird der Befehl `\multirow` definiert, der das Zusammenfassen von mehreren Zeilen in einer Spalte ermöglicht – ähnlich zum Makro `\multicolumn` für Spalten.

## tabularborder

Bei Tabellen wird zwischen Spalten automatisch ein horizontaler Abstand (`\tabcolsep`) gesetzt – besser gesagt jeweils vor und nach einer Spalte. Dies geschieht auch *vor* der ersten und *nach* der letzten Spalte. Dieser zusätzliche Platz an den äußeren Rändern kann störend wirken, insbesondere wenn die Tabelle über die komplette Textbreite gesetzt wird. Mit dem Paket **tabularborder** kann dieser Platz automatisch entfernt werden.

Dies funktioniert allerdings nur mit der `tabular`-Umgebung. Die Tabellen aus den Paketen **tabularx**, **tabulary** und **tabu** werden nicht unterstützt. Wie dieser Abstand bei diesen manuell entfernt werden kann, ist unter Unterkapitel 9.9 zu finden.

v2.02

## tabu

Dies ist ein relativ neues Paket, welches versucht, viele der zuvor genannten Funktionalitäten zu implementieren und weitere bereitzustellen. Dafür werden die Umgebungen `tabu` und `longtabu` definiert. Es kann alternativ zu **tabularx** verwendet werden und ist insbesondere als Ersatz für das Paket **ltxtable** empfehlenswert.

Achtung!

*Für das Paket in der Version v2.8 wären seit einigen Jahren mehrere Bugfixes notwendig. Der originäre Autor hatte außerdem die Änderung der Benutzerschnittstelle in einer zukünftigen Version<sup>54</sup> angekündigt, anscheinend aber mittlerweile die Pflege des Paketes komplett eingestellt.*

Durch Änderungen am  $\text{\LaTeX} 2_{\epsilon}$ -Kernel wurde das Paket unbrauchbar. Deshalb sind durch das  $\text{\LaTeX} 3$ -Projektteam die allernötigsten Anpassungen<sup>55</sup> vorgenommen worden, damit das Paket in der aktuellen Version v2.9 zumindest grundsätzlich lauffähig ist. Ein Großteil der ursprünglich sehr guten Funktionalitäten kann momentan allerdings nicht genutzt werden. Deshalb ist das Paket aktuell nur bedingt empfehlenswert.

## Grafiken und Abbildungen

Grafiken für wissenschaftliche Arbeiten sollten als Vektorgrafiken erstellt werden, um die Skalierbarkeit und hohe Druckqualität zu gewährleisten. Bestenfalls folgen diese auch dem Stil der dazugehörigen Arbeit.<sup>56</sup> Für das Erstellen eigener Vektorgrafiken, welche die  $\text{\LaTeX}$ -Schriften und das Layout des Hauptdokumentes nutzen, gibt es zwei mögliche Ansätze. Entweder die Grafiken werden ähnlich wie das Dokument „programmiert“ oder Zeichenprogramme, welche wiederum die Ausgabe oder das Weiterreichen von Text an  $\text{\LaTeX}$  unterstützen, werden genutzt. Für das Programmieren von Grafiken sollen hier die wichtigsten Pakete vorgestellt werden. Wie diese zu verwenden sind, ist den dazugehörigen Paketdokumentationen zu entnehmen.

### tikz

Dies ist ein sehr mächtiges Paket für das Programmieren von Vektorgrafiken und sehr häufig – insbesondere bei Einsteigern – die erste Wahl bei der Verwendung von **pdf $\text{\LaTeX}$** .

### pstricks

Diese Paket stellt die zweite Variante zum Programmieren von Grafiken dar. Da hiermit auf PostScript (direkt) zugegriffen werden kann, existieren *noch* mehr Möglichkeiten bei der Erstellung eigener Grafiken, wovon die bereitgestellten Befehle rege Gebrauch machen.

<sup>54</sup><https://groups.google.com/d/topic/comp.text.tex/xRGJTC74uCI>

<sup>55</sup><https://github.com/tabu-fixed/tabu>

<sup>56</sup>Für qualitativ hochwertige Dokumente sollten übernommene Grafiken nicht direkt kopiert oder gescannt sondern im gewünschten Zielformat neu erstellt und mit einer Referenz auf die Quelle eingebunden werden.

Daraus resultiert allerdings der Nachteil, dass die mit **pstricks** erstellten Grafiken nicht direkt in eine PDF-Datei kompiliert werden können. Vielmehr müssen die Grafiken aus den Umgebungen `pspicture` zunächst über den Pfad `latex > dvips > ps2pdf` in PDF-Dateien gewandelt werden. Diese lassen sich von **pdfL<sup>A</sup>T<sub>E</sub>X** anschließend als Abbildungen einbinden. Um dieses Vorgehen zu ermöglichen, lassen sich folgende Pakete nutzen:

#### **pst-pdf**

Dieses Paket stellt Methoden für den Export von PostScript-Grafiken in PDF-Dateien bereit. Die einzelnen Aufrufe zur Kompilierung von DVI über PostScript zu PDF müssen durch den Anwender manuell beziehungsweise über die Ausgaberoutinen des verwendeten Editors durchgeführt werden.

#### **auto-pst-pdf, pdftricks2**

Das Paket automatisiert die Erzeugung der **pstricks**-Grafiken mit dem Paket **pst-pdf**. Hierfür muss **pdfL<sup>A</sup>T<sub>E</sub>X** per Option mit erweiterten Schreibrechten ausgeführt werden. Dazu ist der Aufruf von `pdf1atex` mit der Option `--shell-escape` respektive `--enable-write18` notwendig. Bitte beachten Sie dazu ebenfalls die Hinweise in Unterkapitel 9.18. Eine Alternative dazu ist das Paket **pdftricks2**.

Im Tutorial **treatise.pdf** wird für **pstricks** und **tikz** jeweils ein Beispiel gegeben. Um bei der Erstellung von Grafiken mit einem der beiden Paketen nicht bei jeder Änderung das komplette Dokument kompilieren zu müssen, können diese in separate Dateien ausgelagert werden. Hierfür sind die beiden Pakete **standalone** oder **subfiles** sehr nützlich.

Für das Zeichnen einer Grafik mit einem Bildbearbeitungsprogramm, welches die Weiterverarbeitung durch L<sup>A</sup>T<sub>E</sub>X erlaubt, möchte ich auf die freien Programme **LaTeXDraw** und **Inkscape** verweisen. Insbesondere das zuletzt genannte Programm ist sehr empfehlenswert.

#### **svg**

Mit diesem Paket können alle notwendigen Schritte zum Einfügen einer SVG-Grafik in ein L<sup>A</sup>T<sub>E</sub>X-Dokument automatisiert durchgeführt werden. In Unterkapitel 9.19 sind weitere Hinweise hierzu zu finden.

## **Querverweise und Lesezeichen**

Für das Erzeugen von Querverweisen auf bestimmte Gliederungsebenen, Tabellen, Abbildungen oder auch Gleichungen muss für diese besagten Elemente zunächst mit `\label{<Label>}` ein *eindeutiges* Label erzeugt werden, auf welches im Dokument entweder mit `\ref` oder nach dem Laden von **hyperref** besser noch mit `\autoref` referenziert werden kann. In Unterkapitel 9.5 sind diesbezüglich weitere Informationen zu finden.

#### **hyperref**

Hiermit können in einem PDF-Dokument Lesezeichen, Querverweise und Hyperlinks erstellt werden. Wird es geladen, sind außerdem die Option `tudbookmarks` sowie der Befehl `\tudbookmark` nutzbar. Das Paket **bookmark** erweitert die Unterstützung nochmals. Beide genannten Pakete sollten – bis auf sehr wenige Ausnahmen wie beispielsweise **glossaries** – als letztes in der Präambel eingebunden werden.

#### **bookmark**

Dieses Paket verbessert und erweitert die von **hyperref** angebotenen Möglichkeiten zur Erstellung von Lesezeichen – auch Outline-Einträge – im PDF-Dokument. Zum Beispiel können Schriftfarbe- und -stil geändert werden.

## varioref

Mit diesem Paket lassen sich sehr gute Verweise auf bestimmte Seiten erzeugen. Insbesondere, wenn der Querverweis auf die aktuelle, die vorhergehende oder nachfolgende sowie im doppelseitigen Satz auf die gegenüberliegende Seite erfolgt, werden passende Textbausteine für diesen verwendet.

## cleveref

Dieses Paket vereint die Vorzüge der automatischen Benennung referenzierter Objekte mit dem Befehl `\autoref` aus dem Paket **hyperref** und der Verwendung von **varioref**.

## Aufteilung des Hauptdokumentes in Unterdateien

Um während des Entwurfes eines Dokumentes die Zeitdauer für das Kompilieren zu verkürzen, kann dieses in Unterdokumente gegliedert werden. Dadurch wird es möglich, nur den momentan bearbeiteten Dokumentteil – respektive die aktuelle **tikz**- oder **pstricks**-Grafik – zu kompilieren. Die meiner Meinung nach besten Pakete für dieses Unterfangen werden folgend vorgestellt.

v2.02

### standalone

Dieses Paket ist für das Erstellen eigenständiger (Unter-)Dokumente gedacht, welche später in ein Hauptdokument eingebunden werden können. Jedes dieser Teildokumente benötigt eine eigene Präambel. Optional lassen sich die Präambeln der Unterdokumente automatisch in ein Hauptdokument einbinden.

v2.02

### subfiles

Dieses Paket wählt einen etwas anderen Ansatz als **standalone**. Es ist von Anfang an dafür gedacht, ein dediziertes Hauptdokument zu verwenden. Die darin mit `\subfiles` eingebundenen Unterdateien nutzen bei der autarken Kompilierung dessen Präambel.

Unabhängig davon, ob Sie eines der beiden Pakete nutzen oder alles in einem Dokument belassen, ist es ratsam, eigens definierte Befehle, Umgebungen und ähnliches in ein separates Paket auszulagern. Dafür müssen Sie lediglich ein leeres  $\LaTeX$ -Dokument erzeugen und es unter **mypreamble.sty** oder einem anderen Namen im gleichen Ordner wie das Hauptdokument speichern. Dann können Sie in dieser Datei ihre Deklarationen ausführen und diese mit `\usepackage{mypreamble}` in das Dokument einbinden. Dies hat den Vorteil, dass das Hauptdokument zum einen übersichtlich bleibt und Sie zum anderen Ihre persönliche Präambel generisch wachsen lassen und für andere Dokumente wiederverwenden können.

## Die kleinen und großen Helfer...

Hier taucht alles auf, was nicht in die vorherigen Kategorien eingeordnet werden konnte.

### marginnote

Mit dem Befehl `\marginpar` lassen sich Randnotizen erzeugen. Diese sind in  $\LaTeX$  spezielle Gleitobjekte, wodurch selbige nicht immer direkt an der ursprünglich intendierten Stelle am Blattrand gesetzt werden. Das Paket **marginnote** stellt den Befehl `\marginnote` für nicht-gleitende Randnotizen zur Verfügung. Eine Alternative dazu ist Paket **mparhack**.

### todonotes

Mit **todonotes** können noch offene Aufgaben in unterschiedlicher Formatierung am Blattrand oder im direkt Fließtext ausgegeben werden. Aus allen Anmerkungen lässt sich eine Liste aller offenen Punkte erzeugen.

## xparse

Dieses mächtige Paket entstammt dem  $\LaTeX$ 3-Projekt und bietet für die Erstellung eigener Befehle und Umgebungen einen alternativen Ansatz zu den bekannten  $\LaTeX$ -Deklarationsbefehlen `\newcommand` und `\newenvironment` sowie deren Derivaten. Mit **xparse** wird es möglich, obligatorische und optionale Argumente an beliebigen Stellen innerhalb des Befehlskonstruktes zu definieren. Auch die Verwendung anderer Zeichen als eckige Klammern für die Spezifizierung eines optionalen Argumentes ist möglich.

v2.02

## xkeyval, keyval, kvoptions, pgfkeys

Das KOMA-Script-Bundle lädt das Paket **keyval**, um Optionen mit einer Schlüssel-Wert-Syntax deklarieren zu können. Zusätzlich wird von TUD-Script das Paket **kvsetkeys** geladen, um auf nicht definierte Schlüssel reagieren zu können. Die Schlüssel-Wert-Syntax kann auch für eigens definierte Makros genutzt werden, um sich das exzessive Verwenden von optionalen Argumenten zu ersparen. Damit wäre folgende Definition möglich:  
`\newcommand\Befehl[<Schlüssel-Wert-Liste>]{<Argument>}`

Das Paket **xkeyval** erweitert insbesondere die Möglichkeiten zur Deklaration unterschiedlicher Typen von Schlüsseln. Sollten die bereits durch TUD-Script geladenen Pakete **keyval** und **kvsetkeys** in ihrer Funktionalität nicht ausreichen, kann dieses Paket verwendet werden. Für die Entwicklung eigener Pakete, deren Optionen das Schlüssel-Wert-Format unterstützen, kann das Paket **scrbase** genutzt werden. Soll aus einem Grund auf KOMA-Script gänzlich verzichtet werden, sind die beiden Pakete **kvoptions** oder **pgfkeys** eine Alternative.

v2.05

## scrfile

Dieses KOMA-Script-Paket erlaubt es, auf das Laden von Klassen oder Paketen direkt davor oder danach zu reagieren, um beispielsweise Paketabhängigkeiten aufzulösen oder nach dem Laden eines bestimmten Paketes gezielt Befehle anzupassen. Mehr dazu ist im KOMA-Script-Handbuch<sup>57</sup> zu finden.

## calc

Normalerweise lassen sich Berechnungen im Dokument lediglich mit Low-Level- $\TeX$ -Primitiven durchführen. Dieses Paket stellt eine einfachere Syntax für Rechenoperationen der vier Grundrechenarten zur Verfügung. Zusätzlich werden neue Befehle zur Bestimmung der Höhe und Breite bestimmter Textauszüge definiert.

v2.02

## mwe, blindtext

Mit dem Paket **mwe** lassen sich sehr einfach Minimalbeispiele erzeugen, die sowohl Blindtexte als auch Abbildungen enthalten sollen. Werden keine Grafiken sondern lediglich Textabschnitte etc. benötigt, ist das Laden von **blindtext** ausreichend.

## filecontents

Dieses Paket erweitert die im  $\LaTeX$ -Kernel definierte Umgebung `filecontents` dahingehend, dass bereits existierende Dateien überschrieben werden können.

## pdfpages

Das Paket ermöglicht die Einbindung von einzelnen oder mehreren PDF-Dateien.

## crop

Hiermit können eine Beschnittzugabe sowie Schnittmarken – beispielsweise für Poster – erzeugt werden. Hierzu ist in Unterkapitel 9.21 ein Beispiel zu finden.

## pagecolor

Mit dem Paket kann die Hintergrundfarbe der Seiten im Dokument geändert werden.

<sup>57</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>



## afterpage

Der Befehl `\afterpage{ . . . }` kann genutzt werden, um den Inhalt aus dessen Argument direkt nach der Ausgabe der aktuellen Seite auszuführen.

## filemod

Wird entweder **pdfL<sup>A</sup>T<sub>E</sub>X** oder **LuaL<sup>A</sup>T<sub>E</sub>X** als Textsatzsystem eingesetzt, können mit diesem Paket das Änderungsdatum zweier Dateien miteinander verglichen und in Abhängigkeit davon definierbare Aktionen ausgeführt werden.

## listings

Dieses Paket eignet sich hervorragend zur Quelltextdokumentation in L<sup>A</sup>T<sub>E</sub>X. Es bietet die Möglichkeit, externe Quelldateien einzulesen und darzustellen sowie die Syntax in Abhängigkeit der verwendeten Programmiersprache hervorzuheben. Zusätzlich lässt sich ein Verzeichnis mit allen eingebundenen sowie direkt im Dokument angegebenen Quelltextauszügen erstellen. Wird **listings** in Dokumenten mit UTF-8-Kodierung verwendet, sollte direkt nach dem Laden des Paketes in der Präambel Folgendes hinzugefügt werden:

```
\lstset{%
  inputencoding=utf8,extendedchars=true,
  literate=%
    {ä}{\a}1 {ö}{\o}1 {ü}{\u}1
    {Ä}{\A}1 {Ö}{\O}1 {Ü}{\U}1
    {ß}{\ss}1 {~}{\textasciitilde}1
    {»}{\guillemetright}1 {«}{\guillemetleft}1
}
```

## coseoul

Mit diesem Paket kann die Struktur der Gliederung relativ erstellt werden. Es wird keine absolute Gliederungsebene (`\chapter`, `\section`) angegeben sondern die Relation zwischen vorheriger und aktueller Ebene (`\levelup`, `\levelstay`, `\leveldown`).

## selinput, inputenc

Die verwendete Eingabekodierung ist standardmäßig auf `utf8` festgelegt, was von (fast) allen L<sup>A</sup>T<sub>E</sub>X-Editoren unterstützt wird. Dies sollte im Normalfall auch nicht geändert werden. Ältere Dokumente sind aber vielleicht noch in einer anderen Eingabekodierung gespeichert. Kann diese nicht geändert werden, so ist es für **pdfL<sup>A</sup>T<sub>E</sub>X** möglich, eine andere – im Editor (Unterkapitel 9.1) vom Anwender eingestellte – Eingabekodierungen zu nutzen. Diese kann mit dem Paket **selinput** (automatisch) für L<sup>A</sup>T<sub>E</sub>X festgelegt werden Mit:

```
\usepackage{selinput}
\SelectInputMappings{adieresis={ä},germandbls={ß}}
```

wird es verwendet. Alternativ dazu lässt sich mit dem Paket **inputenc** die Eingabekodierung manuell mit `\usepackage[<Eingabekodierung>]{inputenc}` einstellen. Das Paket **fontenc** sollte in jedem Fall *zuvor* geladen werden.

## Bugfixes

### scrhack

Das Paket behebt Kompatibilitätsprobleme der KOMA-Script-Klassen mit den Paketen **hyperref**, **float**, **floatrow** und **listings**. Es ist durchaus empfehlenswert, jedoch sollte unbedingt die Dokumentation beachtet werden.

### scrwfile, morewrites

Falls der Fehler `No room for a new \write` erscheint, kann dieser möglicherweise mit einem dieser beiden Pakete behoben werden. Mehr dazu in Unterkapitel 9.25.

v2.02

### **mparhack**

Zur Behebung falsch gesetzter Randnotizen wird ein Bugfix für `\marginpar` bereitgestellt. Alternativ dazu lässt sich auch **marginnote** verwenden.

### **fix-cm**

Sollte bei einer Schriftgrößenänderung eine oder mehrere Warnungen der Form

`Font shape 'T1/cmr/m/n' in size <...> not available`

erscheinen, so sollte das Paket **fix-cm** vor der Klasse geladen werden. Siehe dazu auch die Hinweise in Unterkapitel 9.24.

## Kapitel 9

# Praktische Tipps & Tricks

## 9.1 $\LaTeX$ -Editoren

Hier werden die gängigsten Editoren zum Erzeugen von  $\LaTeX$ -Dateien genannt. Ich persönlich bin mittlerweile sehr überzeugter Nutzer von *TeXstudio*, da dieser viele Unterstützungs- und Assistenzfunktionen bietet. Neben diesen gibt es noch weitere, gut nutzbare  $\LaTeX$ -Editoren. Unabhängig von der Auswahl des Editors, sollte dieser auf jeden Fall eine Unicode-Unterstützung (UTF-8) enthalten:

- *TeXmaker*
- *Kile*
- *TeXworks*
- *TeXlipse* – Plug-in für *Eclipse*
- *TeXnicCenter*
- *WinEdt*
- *LEd* – früherer  $\LaTeX$  Editor
- *LyX* – grafisches Front-End für  $\LaTeX$

Für *TeXstudio* wird im GitHub-Repository [tud-cd/tudscr](https://github.com/tud-cd/tudscr) das Archiv [tudscr4texstudio.zip](https://github.com/tud-cd/tudscr/releases/download/TeXstudio/tudscr4texstudio.zip)<sup>58</sup> bereitgestellt, welches Dateien zur Erweiterung der automatischen Befehlsvervollständigung für TUD-Script enthält. Diese müssen unter Windows in `%APPDATA%\texstudio` beziehungsweise unter unixoiden Betriebssystemen in `.config/texstudio` eingefügt werden.

Möchten Sie das grafische  $\LaTeX$ -Frontend *LyX* für das Erstellen eines Dokumentes mit den TUD-Script-Klassen nutzen, so werden dafür spezielle Layout-Dateien benötigt, um die Klaskendateien verwenden zu können. Diese sind zusammen mit einem *LyX*-Dokument als Archiv [tudscr4lyx.zip](https://github.com/tud-cd/tudscr/releases/download/LyX/tudscr4lyx.zip)<sup>59</sup> im GitHub-Repository [tud-cd/tudscr](https://github.com/tud-cd/tudscr) verfügbar. Die Layout-Dateien müssen dafür im *LyX*-Installationspfad in den passenden Unterordner kopiert werden. Dieser ist bei Windows `%PROGRAMFILES(X86)%\LyX 2.1\Resources\layouts` beziehungsweise bei unixoiden Betriebssystemen `/usr/share/lyx/layouts`. Anschließend muss *LyX* über den Menüpunkt *Werkzeuge* neu konfiguriert werden.

## 9.2 Literaturverwaltung in $\LaTeX$

v2.02

Die simpelste Variante, eine  $\LaTeX$ -Literaturdatenbank zu verwalten, ist dies mit dem Editor manuell zu erledigen. Wesentlich komfortabler ist es jedoch, die Referenzverwaltung mit einer darauf spezialisierten Anwendung zu bewerkstelligen. Dafür gibt es zwei sehr gute Programme:

- *Citavi*
- *JabRef*

Das Programm *Citavi* ermöglicht den Import von bibliografischen Informationen aus dem Internet. Allerdings sind diese teilweise unvollständig oder mangelhaft. Mit *JabRef* hingegen muss die Literaturdatenbank manuell erstellt werden. Allerdings lassen sich einzelne Einträge aus `.bib`-Dateien importieren. Beide Anwendungen unterstützen den Export beziehungsweise die Erstellung von Datenbanken im Stil von `biblatex`. Für *JabRef* muss diese durch den Anwender explizit aktiviert werden.<sup>60</sup> Zur Verwendung der beiden Programme in Verbindung mit `biblatex` und *biber* gibt es ein gutes Tutorial unter diesem Link.

<sup>58</sup><https://github.com/tud-cd/tudscr/releases/download/TeXstudio/tudscr4texstudio.zip>

<sup>59</sup><https://github.com/tud-cd/tudscr/releases/download/LyX/tudscr4lyx.zip>

<sup>60</sup>Optionen/Einstellungen/Erweitert/BibLaTeX-Modus

## 9.3 Worttrennungen in deutschsprachigen Texten

v2.02

Die möglichen Trennstellen von Wörtern werden von  $\text{\LaTeX}2_{\epsilon}$  mithilfe eines Algorithmus berechnet. Dieser ist jedoch in seiner ursprünglichen Form für die englische Sprache konzipiert worden. Für deutschsprachige Texte wird die Worttrennung – insbesondere bei zusammengesetzten Wörtern – mit dem Paket **hyphsubst** entscheidend verbessert. Dafür wird ein um vielerlei Trennungsmuster ergänztes Wörterbuch aus dem Paket **dehyph-exptl** genutzt.

In jedem Fall wird ein Sprachpaket wie beispielsweise **babel** oder **polyglossia** benötigt, wobei letzteres für den Einsatz mit **Lua $\text{\LaTeX}$**  oder **X $\text{\LaTeX}$**  konzipiert ist. Bei der Verwendung einer dieser beiden Unicode-Textsatzsysteme – in Verbindung mit einem der zwei genannten Sprachpakete werden – die Trennmuster automatisch aktiviert.

Für **pdf $\text{\LaTeX}$**  lassen sich diese mit dem Paket **hyphsubst** aktivieren. Damit auch Wörter mit Umlauten richtig getrennt werden, ist zusätzlich die Verwendung des Paketes **fontenc** mit beispielsweise der T1-Schriftkodierung erforderlich. Der Beginn einer Dokumentpräambel könnte folgendermaßen aussehen:

```
\documentclass[ngerman,<Klassenoptionen>]{<Dokumentklasse>}
\iftutex
  \usepackage{fontspec}
\else
  \usepackage[T1]{fontenc}
  \usepackage[ngerman=ngerman-x-latest]{hyphsubst}
\fi
\usepackage{babel}
. . .
```

Eine Anmerkung noch zur Trennung von Wörtern mit Bindestrichen. Normalerweise sind die beiden von  $\text{\LaTeX}2_{\epsilon}$  verwendeten Zeichen für Bindestrich und Trennstrich identisch. Leider wird der Trennungsalgorithmus von  $\text{\LaTeX}2_{\epsilon}$  bei Wörtern, welche bereits einen Bindestrich enthalten, außer Kraft gesetzt. In der Folge werden – in der deutschen Sprache durchaus öfter anzutreffende – Wortungetüme wie die „Donaudampfschiffahrts-Gesellschafterversammlung“ normalerweise nur direkt nach dem angegebenen Bindestrich getrennt.

Allerdings gibt es die Möglichkeit, das genutzte Zeichen für den Trennstrich zu ändern. Dafür ist das Laden der T1-Schriftkodierung mit dem Paket **fontenc** zwingend erforderlich. Wenn von der verwendeten Schrift nichts anderes eingestellt ist, liegen sowohl Binde- als auch Trennstrich auf Position 45 der Zeichentabelle. In der T1-Schriftkodierung befindet sich auf der Position 127 glücklicherweise für gewöhnlich das gleiche Zeichen noch einmal. Dies ist jedoch von der verwendeten Schrift abhängig. Wird der Ausdruck `\defaultthyphenchar=127` in der Dokumentpräambel verwendet, kann dieses Zeichen für den Trennstrich genutzt werden. Bei den Schriften des Corporate Designs der Technischen Universität Dresden ist dies bereits automatisch eingestellt.

Sollte trotz aller Maßnahmen dennoch einmal ein bestimmtes Wort falsch getrennt werden, so kann die Worttrennung dieses Wortes manuell und global geändert werden. Dies wird mit `\hyphenation{Sil-ben-tren-nung}` gemacht. Es ist zu beachten, dass dies für alle Flexionsformen des Wortes erfolgen sollte. Für eine lokale/temporäre Worttrennung kann mit Befehlen aus dem Paket **babel** gearbeitet werden. Diese sind:

Beschreibung	Befehl
ausschließliche Trennstellen	\-
zusätzliche Trennstellen	" -
Umbruch ohne Trennstrich	" "
Bindestrich ohne Umbruch	"~
Bindestrich, der weitere Trennstellen erlaubt	"=

## 9.4 Lokale Änderungen von Befehlen und Einstellungen

v2.02

Ein zentraler Bestandteil von  $\LaTeX$  ist die Verwendung von Gruppen oder Gruppierungen. Innerhalb dieser bleiben alle vorgenommenen Änderungen an Befehlen, Umgebungen oder Einstellungen lokal. Dies kann sehr nützlich sein, wenn beispielsweise das Verhalten eines bestimmten Makros einmalig oder innerhalb von selbst definierten Befehlen oder Umgebungen geändert werden, im Normalfall jedoch die ursprüngliche Funktionalität behalten soll.

**Beispiel:** Der Befehl `\emph` wird von  $\LaTeX$  für Hervorhebungen im Text bereitgestellt und führt normalerweise zu einer kursiven oder – falls kein Schriftschnitt mit echten Kursiven vorhanden ist – kursivierten oder auch geneigten Auszeichnung. Soll nun in einem bestimmten Abschnitt die Auszeichnung mit fetter Schrift erfolgen, kann der Befehl `\emph` innerhalb einer Gruppierung geändert und verändert werden. Wird diese beendet, verhält sich der Befehl wie gewohnt.

In diesem Text wird genau ein `\emph{Wort}` hervorgehoben.

```
\begingroup
  \renewcommand*{\emph}[1]{\textbf{#1}}%
  In diesem Text wird genau ein \emph{Wort} hervorgehoben.
\endgroup
```

In diesem Text wird genau ein `\emph{Wort}` hervorgehoben.

Eine Gruppierung kann entweder mit `\begingroup` und `\endgroup` oder einfach mit einem geschweiften Klammerpaar `{ . . }` definiert werden.

## 9.5 Bezeichnung der Gliederungsebenen durch hyperref

v2.02

Das Paket `hyperref` stellt für Querverweise unter anderem den Befehl `\autoref{<label>}` zur Verfügung. Mit diesem wird – im Gegensatz zur Verwendung von `\ref` – bei einer Referenz nicht nur die Nummerierung selber sondern auch das entsprechende Element wie Kapitel oder Abbildung vorangestellt. Zur Benennung des referenzierten Elementes wird sequentiell geprüft, ob das Makro `\<Element>autorefname` oder `\<Element>name` existiert. Soll die Bezeichnung eines Elementes geändert werden, muss der entsprechende Bezeichner angepasst werden.

**Beispiel:** Bezeichnungen von Gliederungsebenen können folgendermaßen verändert werden.

```
\renewcaptionname{ngerman}{\sectionautorefname}{Unterkapitel}
\renewcaptionname{ngerman}{\subsectionautorefname}{Abschnitt}
```

## 9.6 URL-Umbrüche im Literaturverzeichnis mit biblatex

v2.02

Wird das Paket **biblatex** verwendet, kann es unter Umständen dazu kommen, dass eine URL nicht vernünftig umbrochen wird. Ist dies der Fall, können die Zähler `biburlnumpenalty`, `biburllucpenalty` und `biburllcpenalty` erhöht werden. Das Manipulieren eines Zähler kann mit `\setcounter{<Zähler>}` oder lokal mit `\defcounter{<Zähler>}` aus dem Paket **etoolbox** erfolgen. Die möglichen Werte liegen zwischen 0 und 10 000, wobei es bei höheren Zählerwerten zu mehr URL-Umbrüchen an Ziffern (`biburlnumpenalty`), Groß- (`biburllucpenalty`) und Kleinbuchstaben (`biburllcpenalty`) kommt. Genaueres in der Dokumentation zu **biblatex**.

## 9.7 Zeilenabstände in Überschriften

Mit dem Paket **setspace** kann der Zeilenabstand beziehungsweise der Durchschuss innerhalb des Dokumentes geändert werden. Sollte dieser erhöht worden sein, können die Abstände bei mehrzeiligen Überschriften als zu groß erscheinen. Um dies zu korrigieren kann mit dem Befehl `\addtokomafont{disposition}{\setstretch{1}}` der Zeilenabstand aller Überschriften auf einzeilig zurückgeschaltet werden. Soll dies nur für eine bestimmte Gliederungsebene erfolgen, so ist der Parameter `disposition` durch das dazugehörige Schriftelement zu ersetzen.

## 9.8 Warnung wegen zu geringer Höhe der Kopf-/Fußzeile

Wird das Paket **setspace** verwendet, kann es passieren, dass nach der Änderung des Zeilenabstandes *innerhalb* des Dokumentes eine oder beide der folgenden Warnungen erscheinen:

```
scrlayer-scrpage Warning: \headheight to low.  
scrlayer-scrpage Warning: \footheight to low.
```

Dies liegt an dem durch den vergrößerten Zeilenabstand erhöhten Bedarf für die Kopf- und Fußzeile, die Höhen können in diesem Fall direkt mit der Verwendung von `\recalctypearea` angepasst werden. Allerdings ändert das den Satzspiegel im Dokument, was eine andere und durchaus berechnete Warnung von **typearea** zur Folge hat. Falls die Änderung des Durchschusses wirklich nötig ist, sollte dies in der Präambel des Dokumentes einmalig passieren. Dann entfallen auch die Warnungen.

## 9.9 Einrückung von Tabellenspalten verhindern

Normalerweise wird in einer Tabelle vor *und* nach jeder Spalte durch  $\LaTeX 2_{\epsilon}$  etwas horizontaler Raum mit `\hskip\tabcolsep` eingefügt.<sup>61</sup> Dies geschieht auch vor der ersten und nach der letzten Spalte. Diese optische Einrückung an den äußeren Rändern kann unter Umständen stören, insbesondere bei Tabellen, die willentlich – beispielsweise mit den Paketen **tabularx**, **tabulary** oder auch **tabu** – über die komplette Seitenbreite aufgespannt werden.

Das Paket **tabularborder** versucht, dieses Problem automatisiert zu beheben, ist jedoch nicht zu allen  $\LaTeX 2_{\epsilon}$ -Paketen für den Tabellensatz kompatibel, unter anderem auch nicht zu den drei zuvor genannten. Allerdings lässt sich dieses Problem manuell lösen.

---

<sup>61</sup>Der Abstand zweier Spalten beträgt folglich `2\tabcolsep`.

Bei der Deklaration einer Tabelle kann mit `@{ . . . }` vor und nach dem Spaltentyp angegeben werden, was anstelle von `\tabcolsep` vor beziehungsweise nach der eigentlichen Spalte eingeführt werden soll. Dies kann für das Entfernen der Einrückungen genutzt werden, indem an den entsprechenden Stellen `@{ }` bei der Angabe der Spaltentypen vor der ersten und nach der letzten Tabellenspalte verwendet wird.

**Beispiel:** Eine Tabelle mit zwei Spalten, wobei bei einer die Breite automatisch berechnet wird, soll über die komplette Textbreite gesetzt werden. Dabei soll der Rand vor der ersten und nach der letzten entfernt werden.

```
\begin{tabularx}{\textwidth}{@{}lX@{}}
. . . & . . . \tabularnewline
. . .
\end{tabularx}
```

## 9.10 Unterdrückung des Einzuges eines Absatzes

Werden zur Absatzauszeichnung im Dokument – wie es aus typografischer Sicht zumeist sinnvoll ist – Einzüge und keine vertikalen Abstände verwendet (KOMA-Script-Option `parskip=false`), kann es vorkommen, dass ein ganz bestimmter Absatz – beispielsweise der nach einer zuvor genutzten Umgebung folgende – ungewollt eingerückt ist. Dies kann sehr einfach manuell behoben werden, indem direkt zu Beginn des Absatzes das Makro `\noindent` aufgerufen wird. Soll das Einrücken von Absätzen nach ganz bestimmten Umgebungen oder Befehlen automatisiert unterbunden werden, ist das Paket `noindentafter` zu empfehlen.

## 9.11 Unterbinden des Zurücksetzens von Fußnoten

Mit `\counterwithin` und `\counterwithout` können  $\LaTeX$ -Zähler so umdefiniert werden, dass sie bei der Änderung eines anderen Zählers zurückgesetzt werden oder nicht. Mit der Angabe von `\counterwithout*{footnote}{chapter}` wird das Zurücksetzen des Fußnotenzählers durch neue Kapitel deaktiviert, womit sich über Kapitel fortlaufende Fußnoten realisieren lassen.

## 9.12 Warnung beim Erzeugen des Inhaltsverzeichnisses

v2.02

Wird mit `\tableofcontents` das Inhaltsverzeichnis für ein Dokument mit einer dreistelligen Seitenanzahl erstellt, so erscheinen unter Umständen viele Warnungen mit der Meldung:

```
overfull \hbox
```

Die Seitenzahlen im Verzeichnis werden in einer Box mit einer festen Breite von `1.55em` gesetzt, welche im Makro `\@pnumwidth` hinterlegt ist und im Zweifel vergrößert werden sollte. Dabei ist auch der rechte Rand für mehrzeilige Einträge im Verzeichnis `\@tocrmarg` zu vergrößern, welcher mit `2.55em` voreingestellt ist. Die Werte sollten nur minimal geändert werden:

```
\makeatletter
\renewcommand*{\@pnumwidth}{1.7em}\renewcommand*{\@tocrmarg}{2.7em}
\makeatother
```



## 9.13 Leer- und Satzzeichen nach $\LaTeX$ -Befehlen

Normalerweise „schluckt“  $\LaTeX$  die Leerzeichen nach einem Makro ohne Argumente. Dies ist jedoch nicht immer – genau genommen in den seltensten Fällen – erwünscht. Für dieses Handbuch ist beispielsweise der Befehl `\TUD` definiert worden, um „Technische Universität Dresden“ nicht ständig ausschreiben zu müssen. Um sich bei der Verwendung des Befehl innerhalb eines Satzes für den Erhalt eines folgenden Leerzeichens das Setzen der geschweiften Klammer nach dem Befehl zu sparen (`\TUD{}`), kann `\xspace` aus dem Paket **xspace** genutzt werden. Damit wird ein folgendes Leerzeichen erhalten. Der Befehl `\TUD` ist wie folgt definiert:

```
\newcommand*\TUD}{Technische Universit\ "at Dresden\xspace}
```

Das Paket **xpunctuate** erweitert die Funktionalität nochmals. Damit können auch Abkürzungen so definiert werden, dass ein versehentlicher Punkt ignoriert wird:

```
\newcommand*\zB}{z. \, B\xperiod}
```

## 9.14 Finden von unbekanntem $\LaTeX$ -Symbolen

Für  $\LaTeX$  stehen jede Menge Symbole zur Verfügung, die allerdings nicht immer einfach zu finden sind. In der Zusammenfassung **symbols-a4.pdf**<sup>62</sup> werden viele Symbole aus mehreren Paketen aufgeführt. Alternativ kann **Detexify**<sup>63</sup> verwendet werden. Auf dieser Webseite wird das gesuchte Symbol einfach gezeichnet, die dazu ähnlichsten werden zurückgegeben.

## 9.15 Das Setzen von Auslassungspunkten

v2.02

Auslassungspunkte werden mit  $\LaTeX 2_{\epsilon}$  mit den Befehlen `\dots` oder `\textellipsis` gesetzt. Für gewöhnlich folgt diesen *immer* ein Leerzeichen, was nicht in jedem Fall gewollt ist. Das Paket **ellipsis** schafft hier Abhilfe, wobei die Option `xspace` führt dazu, dass nach der Verwendung eines der beiden Befehle automatisch ein Leerzeichen gesetzt wird.

```
\usepackage[xspace]{ellipsis}
```

Im Ursprung ist es für das Setzen englischsprachiger Texte gedacht, wo zwischen Auslassungspunkten und Satzzeichen ein Leerzeichen gesetzt wird. Im Deutschen ist dies anders:

„Um eine Auslassung in einem Text zu kennzeichnen, werden drei Punkte gesetzt. Vor und nach den Auslassungspunkten wird jeweils ein Wortzwischenraum gesetzt, wenn sie für ein selbständiges Wort oder mehrere Wörter stehen. Bei Auslassung eines Wortteils werden sie unmittelbar an den Rest des Wortes angeschlossen. Am Satzende wird kein zusätzlicher Schlusspunkt gesetzt. Satzzeichen werden ohne Zwischenraum angeschlossen.“ [Duden, 23. Aufl.]

Um dieses Verhalten zu erreichen, sollte noch Folgendes in der Präambel eingefügt werden:

```
\let\ellipsis punctuation\relax
\newcommand*\qdots}{[\dots{ }]\xspace}
```

Der Befehl `\qdots` wird definiert, um Auslassungspunkte in eckigen Klammern (`[ . . . ]`) setzen zu können, wie sie für das Kürzen von wörtlichen Zitaten häufig verwendet werden.

<sup>62</sup><http://mirrors.ctan.org/info/symbols/comprehensive/symbols-a4.pdf>

<sup>63</sup><http://detexify.kirelabs.org/classify.html>

## 9.16 Lokalisierung für das Setzen von Einheiten mit siunitx

Wenn `siunitx` in einem deutschsprachigen Dokument genutzt soll werden, muss zumindest die richtige Lokalisierung mit `\sisetup{locale = DE}` angegeben werden. Sollen auch die Zahlen richtig formatiert sein, müssen weitere Einstellungen vorgenommen werden. Die meiner Meinung nach besten sind die folgenden.

```
\sisetup{%
  locale = DE,%
  input-decimal-markers={,},input-ignore={.},%
  group-separator={\,},group-minimum-digits=3%
}
```

Das Komma kommt als Dezimaltrennzeichen zum Einsatz. Des Weiteren werden Punkte innerhalb der Zahlen ignoriert und eine Gruppierung von jeweils drei Ziffern vorgenommen. Alternativ zu diesem Paket kann übrigens auch `units` verwendet werden.

## 9.17 Fehlermeldung beim Laden eines Paketes mit Optionen

v2.05

Es kann unter Umständen passieren, dass beim Laden eines Paketes mit bestimmten Optionen via `\usepackage[<Paketoptionen>]{<Paket>}` folgender Fehler ausgegeben wird:

```
! LaTeX Error: Option clash for package <...>.
```

Mit großer Sicherheit wird das angeforderte Paket bereits durch die verwendete Dokumentklasse oder ein anderes Paket geladen. Normalerweise genügt es, die gewünschten Optionen durch `\PassOptionsToPackage{<Paketoptionen>}{<Paket>}` bereits vor dem Laden der Dokumentklasse mit `\documentclass` an das betreffende Paket weiterzureichen.

## 9.18 Probleme bei der Verwendung von auto-pst-pdf

v2.02

Bei der Verwendung von `pdfLATEX` liest das Paket `auto-pst-pdf` die Präambel ein und erstellt anschließend über den PostScript-Pfad `latex > dvips > ps2pdf` eine PDF-Datei, welche lediglich alle in den vorhandenen `pspicture`-Umgebungen erstellten Grafiken enthält. Das Paket `ifpdf` stellt das Makro `\ifpdf` bereit, mit welchem unterschieden werden kann, ob `pdfLATEX` als Textsatzsystem verwendet wird. Abhängig davon können unterschiedliche Quelltexte ausgeführt werden, was genutzt wird, um die nachfolgend beschriebenen Probleme zu beheben.

```
\usepackage{ifpdf}
```

### Die gleichzeitige Verwendung von floatrow

Das Paket `floatrow` stellt Befehle bereit, mit denen die Beschriftung von Gleitobjekten sehr bequem gesetzt werden können. Diese Setzen ihren Inhalt erst in einer Box, um deren Breite zu ermitteln und diese anschließend auszugeben. In Kombination mit `auto-pst-pdf` führt das zu einer doppelten Erstellung der gewünschten Abbildung. Um dies zu vermeiden, müssen die durch `floatrow` bereitgestellten Befehle „unschädlich“ gemacht werden. Die fraglichen Befehlen akzeptieren allerdings bis zu drei optionale Argumente vor den beiden obligatorischen, was für die Benutzerschnittstelle für die (Re-)Definition durch `LATEX 2ε` normalerweise nicht vorgesehen ist. Deshalb wird das Paket `xparse` geladen, mit welchem dies möglich wird. Genaueres dazu ist

der dazugehörigen Paketdokumentation zu entnehmen. Mit folgendem Quelltextauszug lassen sich die **floatrow**-Befehle zusammen mit der `pspicture`-Umgebung wie gewohnt verwenden.

```
\usepackage{floatrow}
\usepackage{xparse}
\ifpdf\else
  \RenewDocumentCommand{\fcapside}{ooo+m+m}{#4#5}
  \RenewDocumentCommand{\ttabbox}{ooo+m+m}{#4#5}
  \RenewDocumentCommand{\ffigbox}{ooo+m+m}{#4#5}
\fi
```

### Die parallele Nutzung von `tikz` und `todonotes`

Mit dem Paket `tikz` – und auch allen anderen Paketen die selbiges nutzen wie beispielsweise `todonotes` – gibt es in Verbindung mit `auto-pst-pdf` ebenfalls Probleme. Lösen lässt sich dieses Dilemma, indem die fraglichen Pakete lediglich geladen werden, wenn `pdf $\LaTeX$`  aktiv ist.

```
\ifpdf
  \usepackage{tikz}%. . .gegebenenfalls weitere auf tikz basierende Pakete
\fi
```

## 9.19 Automatisiertes Einbinden von *Inkscape*-Grafiken

v2.05

Das Einbinden von *Inkscape*-Grafiken in  $\LaTeX$ -Dokumente wird auf Comprehensive TeX Archive Network (CTAN `svg-inkscape`)<sup>64</sup> erläutert. Ein daraus abgeleiteter und verbesserter Ansatz wird durch das Paket `svg` bereitgestellt. Mit diesem Paket ist ein **automatisierter** Export der *Inkscape*-Grafiken direkt bei der Kompilierung mit  $\LaTeX$  und dem anschließenden Einbinden in das Dokument möglich. Hierfür wird der Befehl `\includesvg[<Parameter>]{<SVG-Datei>}` durch das Paket definiert.

Dabei erfolgt der externe Aufruf von *Inkscape* über Kommandozeile beziehungsweise Terminal mit `inkscape.exe`. Damit dieser auch tatsächlich durchgeführt wird, ist die Ausführung einer  $\LaTeX$ -Engine mit der Option `--shell-escape` beziehungsweise `--enable-write18` zwingend notwendig. Außerdem muss der Pfad zur Datei `inkscape.exe` dem System bekannt sein.<sup>65</sup> Für weiterführende Informationen sei auf die Dokumentation des Pakets `svg` verwiesen.

## 9.20 Änderung des Papierformates

Es kann vorkommen, dass innerhalb eines Dokumentes kurzzeitig das Papierformat geändert werden soll, um beispielsweise eine Konstruktionsskizze in der digitalen PDF-Datei einzubinden. Dabei ist es mit der KOMA-Script-Option `paper=<Einstellung>` sowohl möglich, lediglich die Ausrichtung in ein Querformat zu ändern, als auch die Größe des Papierformates selber.

**Beispiel:** Ein Dokument im A4-Format soll kurzzeitig auf ein A3-Querformat geändert werden. Das folgende Minimalbeispiel zeigt, wie dies mit KOMA-Script-Mitteln über die Optionen `paper=landscape` und `paper=A3` geändert werden kann.

```
\documentclass[paper=a4,pagesize]{tudscrreprt}
\usepackage[T1]{fontenc}
```

<sup>64</sup><http://www.ctan.org/pkg/svg-inkscape>

<sup>65</sup>Der Pfad zu `inkscape.exe` in der Umgebungsvariable `PATH` des Betriebssystems enthalten sein.

```

\usepackage[ngerman]{babel}
\usepackage{blindtext}

\begin{document}
\chapter{Überschrift Eins}
\Blindtext

\cleardoublepage
\storeareas\PotraitArea% speichert den aktuellen Satzspiegel
\KOMAOptions{paper=A3,paper=landscape,DIV=current}
\chapter{Überschrift Zwei}
\Blindtext

\cleardoublepage
\PotraitArea% lädt den gespeicherten Satzspiegel
\chapter{Überschrift Drei}
\Blindtext
\end{document}

```

## 9.21 Beschnittzugabe und Schnittmarken

v2.05

Beim Plotten von Postern oder anderen farbigen Druckerzeugnissen besteht oftmals das Problem, dass ein randloses Drucken nur schwer realisierbar ist. Deshalb wird zu oftmals damit beholfen, dass der Druck des fertigen Dokumentes auf einem größeren Papierbogen erfolgt und anschließend auf das gewünschte Zielformat zugeschnitten wird, womit das Problem des nicht bedruckbaren Randes entfällt. Dies kann über zwei verschiedene Wege realisiert werden.

Der einfachste Weg ist die Verwendung des Paketes **crop**. Mit diesem kann das Dokument ganz normal im gewünschten Zielformat erstellt werden. Vor dem Druck wird dieses Paket geladen und einfach das gewünschte Format des Papierbogens angegeben.

```

\RequirePackage{fix-cm}
\documentclass[%
  paper=a1,
  fontsize=36pt
]{tudscrposter}
\usepackage[T1]{fontenc}
. . .
\usepackage{graphicx}
\usepackage[b1,center,cam]{crop}
\begin{document}
. . .
\end{document}

```

Alternativ dazu kann für die TUD-Script-Klassen auf die Funktionalität des Paketes **geometry** zurückgegriffen werden. Dieses Paket stellt den Befehl `\geometry` bereit, in dessen Argument mit `paper=<Papierformat>` das Papierformat festgelegt werden kann. Wird zusätzlich noch der Parameter `layout=<Zielformat>` angegeben, so wird damit das gewünschte Zielformat definiert. Dabei sollte mit `layoutoffset=<Längenwert>` dieser Bereich gegebenenfalls etwas eingerückt werden. Die Angabe von `showcrop=<Ein-Aus-Wert>` generiert außerdem noch visuelle Schnittmarken.

```

\RequirePackage{fix-cm}
\documentclass[%
  paper=a1,
  fontsize=36pt
]{tudscrposter}
\usepackage[T1]{fontenc}
. . .
\geometry{paper=b1,layout=a1,layoutoffset=1in,showcrop}
\begin{document}
. . .
\end{document}

```

Für genauere Erläuterungen sowie weitere Einstellmöglichkeiten sei auf die Dokumentation von **crop** beziehungsweise **geometry** verwiesen. Mit der TUD-Script-Option `bleedmargin` können zusätzlich ie farbigen Bereiche der `tudheadings`-Seitenstile erweitert werden, um ein „Zuschneiden in die Farbe“ zu ermöglichen.

## 9.22 Vermeiden des Skalierens einer PDF-Datei beim Druck

v2.04

Beim Erzeugen eines Druckauftrages einer PDF-Datei kann es unter Umständen dazu führen, dass diese durch den verwendeten PDF-Betrachter unnötigerweise vorher skaliert wird und dabei die Seitenränder vergrößert werden. Um dieses Verhalten für Dokumente, die mit **pdfL<sup>A</sup>T<sub>E</sub>X** erzeugt werden, zu unterdrücken, gibt es zwei Möglichkeiten:

1. Wenn im Dokument ohnehin das Paket **hyperref** verwendet wird, ist der simple Aufruf von `\hypersetup{pdfprintscaling=None}` ausreichend.
2. Der Low-Level-Befehl `\pdfcatalog{/ViewerPreferences<</PrintScaling/None>>}` besitzt das gleiche Verhalten und lässt sich auch ohne **hyperref** nutzen.

Weitere Informationen sind unter <http://www.komascript.de/node/1897> zu finden.

## 9.23 Platzierung von Gleitobjekten

Mit den beiden Paketen **flafter** sowie **placeins** gibt es die Möglichkeit, den für L<sup>A</sup>T<sub>E</sub>X zur Verfügung stehenden Raum für die Platzierung von Gleitobjekten einzuschränken. Darüber hinaus kann diese auch durch die im Folgenden aufgezählten Befehle beeinflusst werden. Die Makros lassen sich mit `\renewcommand*{<Makro>}{<Wert>}` sehr einfach ändern.

<code>\floatpagefraction</code>	Voreinstellung: 0.5
<code>\dblfloatpagefraction</code>	Voreinstellung: 0.5

Der Wert gibt die relative Größe eines Gleitobjektes bezogen auf die eingestellte Texthöhe (`\textheight`) an, die mindestens erreicht sein muss, damit für dieses gegebenenfalls vor dem Beginn eines neuen Kapitels eine separate Seite erzeugt wird. Dabei wird einspaltiges (`\floatpagefraction`) und zweispaltiges (`\dblfloatpagefraction`) Layout unterschieden. Der Wert für beide Befehle sollte im Bereich von 0.5...0.8 liegen.

<code>\topfraction</code>	Voreinstellung: 0.7
<code>\dbltopfraction</code>	Voreinstellung: 0.7

Diese Werte geben den maximalen Seitenanteil für Gleitobjekte, die am oberen Seitenrand platziert werden, für einspaltiges und zweispaltiges Layout an. Er sollte im Bereich von 0.5...0.8

liegen und größer als `\floatpagefraction` beziehungsweise `\dblfloatpagefraction` sein.

`\bottomfraction` Voreinstellung: 0.3

Dies ist der maximale Seitenanteil für Gleitobjekte, die am unteren Seitenrand platziert werden. Er sollte zwischen 0.2 und 0.5 betragen.

`\textfraction` Voreinstellung: 0.2

Dies ist der Mindestanteil an Fließtext, der auf einer Seite mit Gleitobjekten vorhanden sein muss, wenn diese nicht auf einer separaten Seite ausgegeben werden. Er sollte in einem Bereich von 0.1...0.3 liegen.

<code>totalnumber</code> (Zähler)	Voreinstellung: 3
<code>topnumber</code> (Zähler)	Voreinstellung: 2
<code>dbltopnumber</code> (Zähler)	Voreinstellung: 2
<code>bottomnumber</code> (Zähler)	Voreinstellung: 1

Außerdem gibt es noch Zähler, welche die maximale Anzahl an Gleitobjekten pro Seite insgesamt (`totalnumber`) sowie am oberen (`topnumber`) und am unteren Seitenrand (`bottomnumber`) sowie im zweispaltigen Satz beide Spalten überspannend (`dbltopnumber`) festlegen. Die Werte können mit `\setcounter{<Zähler>}{<Wert>}` geändert werden.

<code>\@fptop</code> (Länge)
<code>\@fpsep</code> (Länge)
<code>\@fpbot</code> (Länge)
<code>\@dblftop</code> (Länge)
<code>\@dblfpsep</code> (Länge)
<code>\@dblfpbot</code> (Länge)

Sind vor Beginn eines Kapitels noch Gleitobjekte verblieben, so werden diese durch  $\text{\LaTeX}$  normalerweise auf einer separaten vertikal zentriert Seite ausgegeben. Dabei bestimmen diese Längen jeweils den Abstand vor dem ersten Gleitobjekt zum oberen Seitenrand (`\@fptop`, `\@dblftop`), zwischen den einzelnen Objekten (`\@fpsep`, `\@dblfpsep`) sowie zum unteren Seitenrand (`\@fpbot`, `\@dblfpbot`). Soll dies nicht geschehen, können die Längen durch den Anwender geändert werden.

**Beispiel:** Alle Gleitobjekte auf einer dafür speziell gesetzten Seite sollen direkt zu Beginn dieser ausgegeben werden. In der Dokumentpräambel lässt sich für dieses Unterfangen Folgendes nutzen:

```
\makeatletter
\setlength{\@fptop}{0pt}
\setlength{\@dblftop}{0pt}% twocolumn
\makeatother
```

## 9.24 Warnung bei der Schriftgrößenwahl

v2.04

Die im Dokument verwendete Schriftgröße kann bei den KOMA-Script-Klassen sehr einfach über die Option `fontsize` eingestellt werden, wobei diese immer als Klassenoption angegeben werden sollte. Bei relativ großen und kleinen Schriftgrößen kann dabei eine Warnung in der Gestalt

LaTeX Font Warning: Font shape ‘...’ in size <xx> not available

aufzutreten. Dies daran, dass zum Zeitpunkt des Ladens einer Klasse immer nach den Computer-Modern-Standardschriften gesucht wird, unabhängig davon, ob im Nachhinein ein anderes Schriftpaket geladen wird. Diese sind de-facto nicht in alle Größen skalierbar. Um die Warnungen zu beseitigen, sollte das Paket **fix-cm** mit `\RequirePackage` vor der Dokumentklasse geladen werden:

```
\RequirePackage{fix-cm}
\documentclass[<Klassenoptionen>]{<Klasse>}
\usepackage[T1]{fontenc}
. . .
\begin{document}
. . .
\end{document}
```

Damit werden die Warnungen behoben.

## 9.25 Fehlermeldung: ! No room for a new \write

v2.02

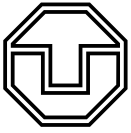
Für das Erstellen und Schreiben externer Hilfsdateien steht  $\LaTeX 2_{\epsilon}$  nur eine begrenzte Anzahl sogenannter Ausgabe-Streams zur Verfügung. Allein für jedes zu erstellende Verzeichnis reserviert  $\LaTeX$  selbst jeweils einen neuen Stream. Auch einige bereits zuvor in diesem Handbuch vorgestellte, sehr hilfreiche Pakete – wie beispielsweise **hyperref**, **biblatex**, **glossaries**, **todonotes** oder auch **filecontents** – benötigen eigene Hilfsdateien und öffnen für das Erstellen dieser einen Ausgabe-Stream oder mehr. Lädt der Anwender mehrere, in eine Hilfsdatei schreibende Pakete, kann es zur Fehlermeldung

```
! No room for a new \write .
```

kommen. Abhilfe schafft hier das Paket **scrwfile**, welches einige Änderungen am  $\LaTeX$ -Kernel vornimmt, um die Anzahl der benötigten Hilfsdateien für das Schreiben aller Verzeichnisse zu reduzieren. Es muss einfach in der Präambel eingebunden werden. Sollten mit diesem Paket unerwarteter Weise Probleme auftreten, ist dessen Anleitung im KOMA-Script-Handbuch<sup>66</sup> zu finden. Eine weitere Möglichkeit, das beschriebene Problem der geringen Menge an Ausgabe-Streams zu umgehen, stellt das Paket **morewrites** dar. Allerdings ist dessen Verwendung nicht in allen Fällen von Erfolg gekrönt.

---

<sup>66</sup><http://mirrors.ctan.org/macros/latex/contrib/koma-script/doc/scrguide.pdf>



## Teil III

# Anhang



Anhang A

## **Weiterführende Installationshinweise**

Achtung!

Hier werden unterschiedliche Varianten erläutert, wie TUD-Script in der Version v2.06o genutzt werden kann, falls eine frühere Variante als **lokale Nutzerinstallation** verwendet wurde.

Bis zur Version v2.01 wurde TUD-Script ausschließlich über das TUD- $\LaTeX$ -Forum<sup>67</sup> zur lokalen Nutzerinstallation angeboten. In erster Linie hat das historische Hintergründe und hängt mit der Entstehungsgeschichte von TUD-Script zusammen. Eine lokale Nutzerinstallation bietet einen – eher zu vernachlässigenden – Vorteil. Treten bei der Verwendung von TUD-Script Probleme auf, können diese im Forum gemeldet und diskutiert werden. Ist für ein solches Problem tatsächlich eine Fehlerkorrektur respektive Aktualisierung von TUD-Script nötig, kann diese schnell und unkompliziert über das GitHub-Repository `tud-cd/tudscr`<sup>68</sup> bereitgestellt und durch den Anwender sofort genutzt werden.

Dies hat allerdings für alle Anwender, welche das Forum relativ wenig oder gar nicht besuchen, den großen Nachteil, dass Sie nicht von Aktualisierungen, Verbesserungen und Fehlerkorrekturen neuer Versionen profitieren können. Auch alle nachfolgenden Bugfixes und Aktualisierungen des TUD-Script-Bundles müssen durch den Anwender manuell durchgeführt werden. Daher wird die Verbreitung via Comprehensive TeX Archive Network (CTAN)<sup>69</sup> präferiert, sodass TUD-Script stets in der gerade aktuellen Version verfügbar ist – eine durch den Anwender aktuell gehaltene  $\LaTeX$ -Distribution vorausgesetzt. Der einzige Nachteil bei diesem Ansatz ist, dass die Verbreitung eines Bugfixes und die anschließende Bereitstellung durch die verwendete Distribution für gewöhnlich bis zu zwei Tagen dauert.

Achtung!

Die gängigen  $\LaTeX$ -Distributionen durchsuchen im Regelfall zuerst das lokale `texmf`-Nutzerverzeichnis nach Klassen und Paketen und erst daran anschließend den `texmf`-Pfad der Distribution selbst. Dabei spielt es keine Rolle, in welchem Pfad die neuere Version einer Klasse oder eines Paketes liegt. Sobald im Nutzerverzeichnis die gesuchte Datei gefunden wurde, wird die Suche beendet. *In der Konsequenz bedeutet dies, dass sämtliche Aktualisierungen über CTAN nicht zum Tragen kommen, falls TUD-Script als lokale Nutzerversion installiert wurde.*

Deshalb wird Anwendern empfohlen, eine gegebenenfalls vorhandene lokale Nutzerinstallation von TUD-Script zu deinstallieren, falls diese nicht *bewusst* installiert wurde. Die Deinstallation wird in Unterkapitel A.1 erläutert. Nach dieser können Updates des TUD-Script-Bundles durch die Aktualisierungsfunktion der Distribution erfolgen.

Wie das TUD-Script-Bundle trotzdem als lokale Nutzerversion installiert oder aktualisiert werden kann, ist in Unterkapitel A.2 beziehungsweise Unterkapitel A.3 zu finden. Der Anwender sollte in diesem Fall allerdings genau wissen, was er damit bezweckt, da er in diesem Fall für die Aktualisierung von TUD-Script selbst verantwortlich ist.

### Nutzung der veralteten Schriftfamilien

v2.06

Soll ein Dokument noch mit den veralteten Schriftfamilien **Univers** und **DIN Bold** gesetzt werden, so ist eine lokale Installation der Type1-Schriften notwendig, welche in Unterkapitel A.4 beschrieben wird. Zusätzlich sei auf die beiden Optionen `tudscrver=2.05` und `cdo1dfont` hingewiesen. Für die Nutzung der **Open Sans** wird lediglich die aktuelle Version des Paketes `opensans` benötigt, welches auf Comprehensive TeX Archive Network (CTAN)<sup>70</sup> zu finden ist.

<sup>67</sup><http://latex.wcms-file3.tu-dresden.de/phpBB3/>

<sup>68</sup><https://github.com/tud-cd/tudscr/releases>

<sup>69</sup><http://www.ctan.org/pkg/tudscr>

<sup>70</sup><http://www.ctan.org/pkg/opensans>

## A.1 Lokale Deinstallation des TUD-Script-Bundles

Über die Kommandozeile beziehungsweise das Terminal kann mit

```
kpsewhich -all tudscrbase.sty
```

überprüft werden, ob eine lokale Nutzerinstallation von TUD-Script vorhanden ist. Es werden alle Pfade ausgegeben, in denen die Datei **tudscrbase.sty** gefunden wird. Erscheint nur der Distributionspfad, ist die TUD-Script-Version von CTAN aktiv und der Anwender kann mit dem TUD-Script-Bundle arbeiten.

Wird *nur* das lokale Nutzerverzeichnis oder gar kein Verzeichnis gefunden, so wird höchstwahrscheinlich eine veraltete Distribution verwendet. In diesem Fall wird eine Aktualisierung dieser *unbedingt* empfohlen. Sollte dies nicht möglich sein, *muss* TUD-Script als lokale Nutzerversion installiert (Unterkapitel A.2) beziehungsweise – falls ein Pfad ausgegeben wurde – aktualisiert (Unterkapitel A.3) werden.

Sollte neben dem Pfad der Distribution noch mindestens ein weiterer Pfad angezeigt werden, so ist eine lokale Nutzerversion installiert. In diesem Fall hat der Anwender drei Möglichkeiten:

1. Entfernen der lokalen Nutzerinstallation (skriptbasiert)
2. Entfernen der lokalen Nutzerinstallation (manuell)
3. Aktualisierung der lokalen Nutzerversion (Unterkapitel A.3)

Um die lokale Nutzerinstallation zu entfernen, kann für Windows **tudscr\_uninstall.bat**<sup>71</sup> sowie für unixartige Betriebssysteme **tudscr\_uninstall.sh**<sup>72</sup> verwendet werden. Nach der Ausführung des jeweiligen Skriptes kann mit dem zu Beginn gezeigten Aufruf in der Kommandozeile respektive Terminal geprüft werden, ob die Deinstallation erfolgreich war. Wird immer noch mindestens ein lokaler Pfad ausgegeben, sollte TUD-Script manuelle deinstalliert werden, was nachfolgend beschrieben wird.

Nur die Deinstallation aller lokalen Nutzerinstallationen von TUD-Script ermöglicht die Verwendung der jeweils aktuellen CTAN-Version. Hierfür ist – unter der Annahme, dass das automatisierte Deinstallieren mithilfe der zuvor genannten Skripte zur Deinstallation nicht erfolgreich war – etwas Handarbeit durch den Anwender vonnöten. Der in der Kommandozeile respektive im Terminal mit

```
kpsewhich -all tudscrbase.sty
```

gefundene – zum Ordner der Distribution *zusätzliche* – Pfad hat die folgende Struktur:

```
<Installationspfad>/tex/latex/tudscr/tudscrbase.sty
```

Um die Nutzerinstallation vollständig zu entfernen, muss als erstes zu *<Installationspfad>* navigiert werden. Anschließend ist in diesem Pfad Folgendes durchzuführen:

```
tex/latex/tudscr/  .cls- und .sty-Dateien löschen
tex/latex/tudscr/  Ordner logo vollständig löschen
doc/latex/         Ordner tudscr vollständig löschen
source/latex/     Ordner tudscr vollständig löschen
```

Zum Abschluss ist in der Kommandozeile beziehungsweise im Terminal der Befehl `texhash` aufzurufen. Damit wurde die lokale Nutzerversion entfernt und es wird von nun an die Version von TUD-Script genutzt, welche durch die verwendete Distribution bereitgestellt wird.

<sup>71</sup>[https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr\\_uninstall.bat](https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr_uninstall.bat)

<sup>72</sup>[https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr\\_uninstall.sh](https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr_uninstall.sh)

## A.2 Lokale Installation des TUD-Script-Bundles

Eine lokale Nutzerinstallation von TUD-Script sollte ausschließlich durch Anwender ausgeführt werden, die genau wissen, aus welchen Gründen dies geschehen soll. Hierfür werden die passenden Skripte angeboten. Für eine lokale Nutzerinstallation des TUD-Script-Bundles unter Windows für die Distributionen  $\text{T}\epsilon\text{X}$  Live oder  $\text{M}\text{i}\text{K}\text{T}\epsilon\text{X}$  werden die Dateien aus `TUD-Script_v2.06o_Windows.zip`<sup>73</sup> benötigt. Für unixoide Betriebssysteme sind es die Dateien aus `TUD-Script_v2.06o_Unix.zip`<sup>74</sup>.

Beim Ausführen des spezifischen Installationskscripts – `tudscr_v2.06o_install.bat` für Windows respektive `tudscr_v2.06o_install.sh` für unixoide Betriebssysteme – werden alle Dateien in das lokale Nutzerverzeichnis der jeweiligen Distribution installiert, falls kein anderes Verzeichnis explizit angegeben wird. *In jedem Fall sollte vor der Ausführung des Installationskscripts ein Update der  $\text{L}\text{T}\epsilon\text{X}$ -Distribution durchgeführt werden. Dies gilt insbesondere bei der Nutzung von  $\text{M}\text{i}\text{K}\text{T}\epsilon\text{X}$ . Andernfalls wird es unter Umständen im Installationsprozess oder bei der Nutzung von TUD-Script zu Problemen kommen.*

Achtung!

Alternativ zur Nutzung der bereitgestellten Skripte kann auch einfach der Inhalt des Archivs `tudscr_v2.06o.zip`<sup>75</sup> in das lokale `texmf`-Nutzerverzeichnis kopiert und abschließend in der Kommandozeile respektive im Terminal `texhash` aufgerufen werden.

## A.3 Lokales Update des TUD-Script-Bundles

### A.3.1 Update des TUD-Script-Bundles ab Version v2.02

Ein Update und die lokale Installation unterscheiden sich ab der Version v2.06 nicht voneinander, das Vorgehen ist absolut identisch zu der Beschreibung in Unterkapitel A.2. *Die lokale Aktualisierung auf Version v2.06o funktioniert allerdings nur, wenn TUD-Script bereits mindestens in der Version v2.02 entweder als lokale Nutzerversion oder über die Distribution installiert ist.*

Achtung!

### A.3.2 Update des TUD-Script-Bundles ab Version v2.00

Mit der Version v2.02 gab es einige tiefgreifende Änderungen. Deshalb wird für vorausgehende Versionen – sprich v2.00 und v2.01 – kein dediziertes Update angeboten. Die Aktualisierung kann durch den Anwender entweder – wie in Unterkapitel A.2 erläutert – mit einer skriptbasierten oder mit einer manuellen Neuinstallation erfolgen.

### A.3.3 Update des TUD-Script-Bundles von Version v1.0

Ist TUD-Script in der veralteten *Version v1.0* installiert, so wird vor der Aktualisierung dringlichst zu einem vollständigen Entfernen dieser Version geraten. Andernfalls werden nach einem Update bei der Verwendung massive Probleme und Fehler auftreten. Zur Deinstallation werden die Skripte `tudscr_uninstall.bat`<sup>76</sup> respektive `tudscr_uninstall.sh`<sup>77</sup> bereitgestellt. Die aktuelle Version v2.06o kann nach der vollständigen Deinstallation aller veralteten Versionen wie in Unterkapitel A.4 beschrieben installiert werden.

<sup>73</sup>[https://github.com/tud-cd/tudscr/releases/download/v2.06o/TUD-Script\\_v2.06o\\_Windows.zip](https://github.com/tud-cd/tudscr/releases/download/v2.06o/TUD-Script_v2.06o_Windows.zip)

<sup>74</sup>[https://github.com/tud-cd/tudscr/releases/download/v2.06o/TUD-Script\\_v2.06o\\_Unix.zip](https://github.com/tud-cd/tudscr/releases/download/v2.06o/TUD-Script_v2.06o_Unix.zip)

<sup>75</sup>[https://github.com/tud-cd/tudscr/releases/download/v2.06o/tudscr\\_v2.06o.zip](https://github.com/tud-cd/tudscr/releases/download/v2.06o/tudscr_v2.06o.zip)

<sup>76</sup>[https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr\\_uninstall.bat](https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr_uninstall.bat)

<sup>77</sup>[https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr\\_uninstall.sh](https://github.com/tud-cd/tudscr/releases/download/uninstall/tudscr_uninstall.sh)

Im Vergleich zur *Version v1.0* hat sich an der Benutzerschnittstelle nicht sehr viel verändert, ein Umstieg auf die Version v2.06o dürfte keine Schwierigkeiten bereiten. Treten danach dennoch Probleme auf, sollte der Anwender als erstes die Beschreibung des Paketes **tudscrcomp** in Unterkapitel B.2 auf Seite 144 lesen, welches eine Schnittstelle zur Nutzung alter und ursprünglich nicht mehr vorgesehener Befehle sowie Optionen bereitstellt. Allerdings werden einige von diesen auch durch das Paket **tudscrcomp** nicht mehr bereitgestellt. Aufgeführt sind diese in Anhang B. Sollten trotz aller Hinweise dennoch Fehler oder Probleme beim Umstieg auf die neue TUD-Script-Version auftreten, ist eine Meldung im TUD- $\LaTeX$ -Forum<sup>78</sup> oder im GitHub-Repository **tud-cd/tudscr**<sup>79</sup> die beste Möglichkeit, um Hilfe zu erhalten.

## A.4 Installation veralteter Schriftfamilien

Bis Anfang des Jahres 2018 nutzte das Corporate Design der Technischen Universität Dresden als Hausschrift nicht **Open Sans** sondern die Schriftfamilien **Univers** und **DIN Bold**. Diese lassen sich auch weiterhin mit TUD-Script verwenden, um alte Dokumente kompilieren zu können. Hierfür sei auf die Optionen `tudscrverv=2.05` und `cdo1dfont` hingewiesen. Da es sich bei diesen um lizenzierte Schriften handelt, müssen diese beim Universitätsmarketing auf Anfrage<sup>80</sup> mit dem Hinweis auf die Verwendung von  $\LaTeX$  bestellt und nach Erhalt der notwendigen Archive **Univers\_PS.zip** und **DIN\_Bd\_PS.zip** für Windows (Abschnitt A.4.1) beziehungsweise unixoide Betriebssysteme (Abschnitt A.4.2) installiert werden. Ohne einen triftigen Grund sollte jedoch in jedem Fall die **Open Sans** genutzt werden, insbesondere für neu erstellte Dokumente.

Das TUD-Script-Bundle unterstützt besagte Schriften auch im OpenType-Format, welche ebenfalls über das Universitätsmarketing auf Anfrage bestellt werden müssen. Die in den Archiven **Univers\_8\_TTF.zip** und **DIN\_TTF.zip** enthaltenen Schriften lassen sich – sobald diese für das Betriebssystem installiert wurden – mit dem Paket **fontspec** verwenden. In Unterkapitel 2.1 auf Seite 15 sind weitere Hinweise zur Verwendung des Paketes **fontspec** zu finden.

Im GitHub-Repository **tud-cd/tudscr** sind die zur Schriftinstallation benötigten Skripte<sup>81</sup> ebenso zu finden wie die Skripte<sup>82</sup> für die Klassen von Klaus Bergmann. Die unterschiedlichen Installationsskripte begründen sich insbesondere dadurch, dass bei der Installation für das TUD-Script-Bundle sowohl die Metriken als auch das Kerning der Schriften für Fließtext und den Mathematikmodus angepasst werden. Sollen die so verbesserten Schriften für die Klassen von Klaus Bergmann verwendet werden, kann dies mit dem Paket **fix-tudscrfonts** erfolgen, was allerdings das Ergebnis der erzeugten Ausgabe beeinflusst, weshalb die Installationsskripte in unterschiedlichen Varianten weiterhin vorgehalten werden.

### Erforderliche Pakete bei der Schriftinstallation

Für die Installation der Schriften sind die Pakete **fontinst**, **cmbright**, **hfbright**, **cm-super** und **iwona** von *essentieller* Bedeutung und daher *zwingend* notwendig. Das Vorhandensein dieser wird durch die jeweiligen Schriftinstallationsskripte geprüft und die Installation beim Fehlen eines oder mehrerer Pakete mit einer entsprechenden Warnung abgebrochen.

<sup>78</sup><http://latex.wcms-file3.tu-dresden.de/phpBB3/>

<sup>79</sup><https://github.com/tud-cd/tudscr/issues>

<sup>80</sup><https://tu-dresden.de/cd/>

<sup>81</sup><https://github.com/tud-cd/tudscr/releases/tag/fonts>

<sup>82</sup><https://github.com/tud-cd/tudscr/releases/tag/oldfonts>

### Anmerkung zu MiKTeX

Vor der Installation der Schriften für TUD-Script sollte unbedingt ein Update von MiKTeX durchgeführt werden. Außerdem ist es sehr ratsam, die Installation von MiKTeX in der Mehrbenutzervariante mit Administratorrechten durchzuführen, da die Einzelbenutzervariante relativ unregelmäßig und nicht immer nachvollziehbar zu Problemen führen kann.

Möglicherweise sind einige der für den Schriftinstallationsprozess notwendigen Pakete noch nicht installiert. Ist die automatische Nachinstallation fehlender Pakete aktiviert, so reicht es im Normalfall das Installationskript zu starten. Andernfalls müssen diese Pakete manuell durch den Benutzer über den MiKTeX-Paketmanager hinzugefügt werden.

Das Installationskript scheitert außerdem bei einigen Anwendern – aufgrund eingeschränkter Nutzerzugriffsrechte – beim Eintragen der Schriften in die Map-Datei. Dies muss gegebenenfalls durch den Anwender über die Kommandozeile

```
initexmf --edit-config-file updmap
```

erfolgen. In der sich öffnenden Datei sollte sich der Eintrag `Map tudscr.map` befinden. Ist dies nicht der Fall, muss diese Zeile manuell eingetragen und die Datei anschließend gespeichert werden. Danach muss der Nutzer in der Kommandozeile noch folgenden Aufruf ausführen:

```
initexmf --mkmaps
```

### Anmerkung zu T<sub>E</sub>X Live und MacT<sub>E</sub>X

Sollte keine Vollinstallation von T<sub>E</sub>X Live durchgeführt worden sein, müssen die erforderlichen Pakete zur Schriftinstallation manuell über den T<sub>E</sub>X Live-Paketmanager hinzugefügt werden.

Sind nach einem fehlerfreien Durchlauf des Installationskriptes die Schriften dennoch nicht verfügbar, so muss die Synchronisierung aller Schriftdateien angestoßen werden. Daran anschließend müssen die Map-Datei und die dazugehörigen Schriftdateien registriert werden. Die hierfür notwendigen Aufrufe lauten:

```
updmap-sys --syncwithtrees  
updmap-sys --enable Map=tudscr.map  
updmap-sys --force
```

Sind die Schriften danach immer noch nicht verfügbar, so wurden bestimmt schon weitere Schriften auf dem System *lokal* installiert. In diesem Fall sollte der Vorgang nochmals für eine lokale Schriftinstallation mit

```
updmap --syncwithtrees  
updmap --enable Map=tudscr.map  
updmap --force
```

ausgeführt werden. Dieses Vorgehen macht allerdings den Befehl `updmap-sys` von nun an wirkungslos. Nach einer systemweiten Installation neuer Schriften – beispielsweise bei der Aktualisierung der Distribution – müssen diese über den manuellen Aufruf von `updmap` zukünftig durch den Anwender lokal bei T<sub>E</sub>X Live respektive MacT<sub>E</sub>X registriert werden.

Achtung!

*Für die Schriftinstallation werden die Skripte `tftopl`, `pltotf` und `vptovf` benötigt, welche bei T<sub>E</sub>X Live beziehungsweise MacT<sub>E</sub>X über das Paket `fontware` aus `collection-fontutils` bereitgestellt werden und zwingend installiert sein müssen.*

## A.4.1 Installation der Type1-Schriften unter Windows

Zur Installation der Schriften des Corporate Designs für das TUD-Script-Bundle ist das Archiv `TUD-Script_fonts_Windows.zip`<sup>83</sup> vorgesehen. Dieses ist sowohl für **T<sub>E</sub>X Live** als auch **MiK<sub>T</sub>E<sub>X</sub>** nutzbar und enthält – bis auf die jeweiligen Schriftarchive selbst – alle benötigten Dateien. Diese sollten nach dem Entpacken des Archivs in das gleiche Verzeichnis kopiert werden. Vor der Verwendung des Skripts `tudscr_fonts_install.bat` sollte sichergestellt werden, dass sich *alle* der folgenden Dateien im selben Verzeichnis befinden:

<code>tudscr_fonts_install.bat</code>	Installationsskript
<code>Univers_PS.zip</code>	Archiv mit Schriftdateien für <b>Univers</b>
<code>DIN_Bd_PS.zip</code>	Archiv mit Schriftdateien für <b>DIN Bold</b>
<code>tudscr_fonts_install.zip</code>	Archiv mit Metriken für die Schriftinstallation via <b>fontinst</b>

Beim Ausführen des Installationsskripts werden alle Schriften standardmäßig in ein lokales Nutzerverzeichnis installiert. Wird das Skript über das Kontextmenü mit Administratorrechten ausgeführt, erfolgt die Installation in einem Pfad, der *für alle Nutzer* gültig und lesbar ist.

## A.4.2 Installation der Type1-Schriften unter Linux und OS X

Zur Installation der Schriften des Corporate Designs für das TUD-Script-Bundle ist das Archiv `TUD-Script_fonts_Unix.zip`<sup>84</sup> vorgesehen. Dieses ist sowohl für **T<sub>E</sub>X Live** als auch **MacT<sub>E</sub>X** nutzbar und enthält – bis auf die jeweiligen Schriftdateien selbst – alle benötigten Dateien. Diese sollten nach dem Entpacken des Archivs in das gleiche Verzeichnis kopiert werden. Vor der Verwendung des Skripts `tudscr_fonts_install.sh` sollte sichergestellt werden, dass sich *alle* der folgenden Dateien im selben Verzeichnis befinden:

<code>tudscr_fonts_install.sh</code>	Installationsskript (Terminal: <code>bash tudscr_fonts_install.sh</code> )
<code>Univers_PS.zip</code>	Archiv mit Schriftdateien für <b>Univers</b>
<code>DIN_Bd_PS.zip</code>	Archiv mit Schriftdateien für <b>DIN Bold</b>
<code>tudscr_fonts_install.zip</code>	Archiv mit Metriken für die Schriftinstallation via <b>fontinst</b>

### Anmerkung zu Linux und OS X

Achtung!

*Nach dem Entpacken eines Release-Archivs im passenden Pfad<sup>85</sup> muss das Skript zwingend mit `bash <Skript>.sh` im Terminal in diesem Pfad mit den benötigten Dateien aufgerufen werden. Dabei werden alle Schriften standardmäßig in das lokale Nutzerverzeichnis (`$TEXMFHOME`) installiert. Wird das Skript mit `sudo` verwendet, erfolgt die Installation *für alle Nutzer* in den lokalen Systempfad (`$TEXMFLOCAL`).*

Es ist unbedingt darauf zu Achten, dass beim Ausführen des Skriptes das Terminal im richtigen Verzeichnis aktiv ist. Bei den meisten unixoiden Betriebssystemen ist es problemlos möglich, das Terminal aus der Benutzeroberfläche heraus über das Kontextmenü im gewünschten Pfad zu öffnen. Geht dies nicht, so muss nach dem Öffnen des Terminals mit dem Befehl `cd` erst zum entsprechenden Pfad – exemplarisch `cd "$HOME/Downloads/<Unterordner>"` – navigiert werden. Ein beispielhafter Aufruf im Terminal könnte also lauten:

```
cd "$HOME/Downloads/TUD-Script_fonts_Unix" [ENTER]
bash tudscr_fonts_install.sh [ENTER]
```

<sup>83</sup>[https://github.com/tud-cd/tudscr/releases/download/fonts/TUD-Script\\_fonts\\_Windows.zip](https://github.com/tud-cd/tudscr/releases/download/fonts/TUD-Script_fonts_Windows.zip)

<sup>84</sup>[https://github.com/tud-cd/tudscr/releases/download/fonts/TUD-Script\\_fonts\\_Unix.zip](https://github.com/tud-cd/tudscr/releases/download/fonts/TUD-Script_fonts_Unix.zip)

<sup>85</sup>beispielsweise `cd "$HOME/Downloads/<Unterordner>"`

### A.4.3 Installationshinweise für portable Installationen

Prinzipiell ist die Installation der Type1-Schriften des Corporate Designs bei der Nutzung von **T<sub>E</sub>X Live Portable** respektive **MiK<sub>T</sub>E<sub>X</sub> Portable** äquivalent zur nicht-portablen Variante, welche in Unterkapitel A.4 beschrieben wird. Alle dort gegebenen Hinweise sollten sorgfältig berücksichtigt werden. Der durch das jeweilige Installationskript voreingestellte Installationspfad sollte für gewöhnlich nicht geändert werden. Geschieht dies dennoch, so sollte dieser sich logischerweise auf dem externen Speichermedium  $\langle \text{Laufwerksbuchstabe} \rangle : \backslash$  befinden.

#### **T<sub>E</sub>X Live Portable**

Das folgende Vorgehen wurde mit Windows getestet. Empfehlungen für die portable Installation für unixoide Betriebssysteme können an [hanisch.latex@outlook.com](mailto:hanisch.latex@outlook.com) gesendet werden.

1. Installation von **T<sub>E</sub>X Live Portable** in  $\langle \text{Laufwerksbuchstabe} \rangle : \backslash \text{LaTeX} \backslash \text{texlive}$
2. Die Datei **tl-tray-menu.exe** im Installationspfad öffnen
3. Das Kontextmenü von **T<sub>E</sub>X Live Portable** mit einem Rechtsklick auf das entsprechende Symbol im Infobereich der Taskleiste öffnen und entweder über die grafische Oberfläche (*Package Manager*) oder die Kommandozeile (*Command Prompt*) ein Update durchführen
4. Über das Kontextmenü die Kommandozeile ausführen und in dieser das Skript für die Installation der Schriften **tudscr\_fonts\_install.bat** starten. Dabei gegebenenfalls zuvor in den Pfad des Skriptes wechseln – exemplarisch:

```
cd /d %USERPROFILE%\Downloads\TUD-Script_fonts_Windows [ENTER]
tudscr_fonts_install.bat [ENTER]
```

Unter Umständen meldet das Skript fehlende Pakete. Dieses müssen über durch den Anwender über den *T<sub>E</sub>X Live Manager* installiert werden. *Ein Ausführen ohne die über T<sub>E</sub>X Live Portable geöffnete Kommandozeile führt zu Fehlern.*

Achtung!

#### **MiK<sub>T</sub>E<sub>X</sub> Portable**

1. Installation von **MiK<sub>T</sub>E<sub>X</sub> Portable** in  $\langle \text{Laufwerksbuchstabe} \rangle : \backslash \text{LaTeX} \backslash \text{MiKTeX}$ <sup>86</sup>
2. Die Datei **miktex-portable.cmd** im Installationspfad öffnen
3. Das Kontextmenü von **MiK<sub>T</sub>E<sub>X</sub> Portable** mit einem Rechtsklick auf das entsprechende Symbol im Infobereich der Taskleiste öffnen und ein Update durchführen
4. Über das Kontextmenü die Kommandozeile ausführen und in dieser das Skript für die Installation der Schriften **tudscr\_fonts\_install.bat** starten. Dabei gegebenenfalls zuvor in den Pfad des Skriptes wechseln – exemplarisch:

```
cd /d %USERPROFILE%\Downloads\TUD-Script_fonts_Windows [ENTER]
tudscr_fonts_install.bat [ENTER]
```

Bei diesem Schritt werden die Pakete **fontinst**, **cmbright** und **iwona** unter Umständen nachinstalliert. *Ein Ausführen ohne die über MiK<sub>T</sub>E<sub>X</sub> Portable geöffnete Kommandozeile führt zu Fehlern.*

Achtung!

5. Bei der erstmaligen Verwendung von TUD-Script werden alle weiteren benötigten Pakete von **MiK<sub>T</sub>E<sub>X</sub> Portable** nachinstalliert, falls die automatische Nachinstallation aktiviert ist und diese nicht schon vorhanden sind. Unter anderem betrifft dies die Pakete **tudscr**, **koma-script** sowie **etoolbox**, **xpatch** **letltxmacro**, **environ**, **trimspaces**, **xcolor** und **mptopdf**.

<sup>86</sup>Der Pfad darf *nicht* auf der obersten Verzeichnisebene  $\langle \text{Laufwerksbuchstabe} \rangle : \backslash$  liegen.



#### A.4.4 Probleme bei der Installation der Type1-Schriften

Wird Windows verwendet, kann es unter Umständen vorkommen, dass notwendige Befehlsaufrufe für das Installationsskript nicht ausgeführt werden können. In diesem Fall ist der Pfad zu den benötigten Dateien, welche normalerweise unter %SystemRoot%\System32 zu finden sind, nicht in der Umgebungsvariable PATH enthalten. Einen Hinweis zur Problemlösung ist in diesem Beitrag im Forum<sup>87</sup> zu finden.

Treten bei der Installation wider Erwarten Probleme auf, so ist zur Lösung eine Logdatei zu erstellen. Hierfür sollte unter **Windows** das Skript, welches Probleme verursacht, *nicht* aus der Kommandozeile oder dem Explorer heraus sondern über *Windows PowerShell* ausgeführt werden. Hierfür ist die Eingabe von „PowerShell“ im Startmenü von Windows mit einem nachfolgenden Öffnen mittels [ENTER]-Taste ausreichend. Danach muss mit `cd` zum Ordner des Skriptes navigiert und dieses mit `.\<Skript>.bat|Tee-Object -file <Skript>.log` ausgeführt werden. Ein Aufruf aus der PowerShell-Konsole könnte lauten:

```
cd "$env:USERPROFILE\Downloads\TUD-Script_fonts_Windows" [ENTER]
.\tudscr_fonts_install.bat|Tee-Object -file fonts_install.log [ENTER]
```

Für **unixartige Systeme** ist der Aufruf `bash <Skript>.sh > <Skript>.log` aus dem Terminal heraus zu verwenden. Ein exemplarische Verwendung könnte lauten:

```
cd "$HOME/Downloads/TUD-Script_fonts_Unix" [ENTER]
bash tudscr_fonts_install.sh > fonts_install.log [ENTER]
```

Die so erstellte Logdatei kann *mit einer kurzen Fehlerbeschreibung* entweder im TUD-~~La~~TeX-Forum gepostet oder per E-Mail an `hanisch.latex@outlook.com` gesendet werden.

---

<sup>87</sup><http://latex.wcms-file3.tu-dresden.de/phpBB3/viewtopic.php?t=359>

## Anhang B

# Obsoleete sowie vollständig entfernte Optionen und Befehle

## B.1 Veraltete Optionen und Befehle in TUD-Script

Einige Optionen und Befehle waren während der Weiterentwicklung von TUD-Script in ihrer ursprünglichen Form nicht mehr umsetzbar oder wurden – unter anderem aus Gründen der Kompatibilität zu anderen Paketen – schlichtweg verworfen. Dennoch besteht für die meisten entfallenen Direktiven eine Möglichkeit, deren Funktionalität ohne größere Aufwände mit TUD-Script in der aktuellen Version v2.06o darzustellen. Ist dies der Fall, wird hier entsprechend kurz darauf hingewiesen.

### B.1.1 Änderungen für TUD-Script v2.00

<code>cd=alternative</code>	(entfällt)
<code>cdtitle=alternative</code>	(entfällt)
<code>\titlecolwidth</code> (Länge)	(entfällt)
<code>\authortext</code> (Bezeichner)	(entfällt)

Die alternative Titelseite ist komplett aus dem TUD-Script-Bundle entfernt worden. Dementsprechend entfallen auch die dazugehörigen Optionen sowie Länge und Bezeichner.

<code>color=&lt;Ein-Aus-Wert&gt;</code>	(siehe <code>cd</code> auf Seite 24)
---	--------------------------------------

Die Einstellungen der farbigen Ausprägung des Dokumentes erfolgt über die Option `cd`.

<code>tudfonts=&lt;Ein-Aus-Wert&gt;</code>	(siehe <code>cdfont</code> auf Seite 16)
--	--

Die Option zur Schrifteinstellung ist wesentlich erweitert worden. Aus Gründen der Konsistenz wurde diese umbenannt.

<code>tudfoot=&lt;Ein-Aus-Wert&gt;</code>	(siehe <code>cdfoot</code> auf Seite 29)
---	--

Ebenso wurde die Option `tudfoot` umbenannt, um dem Namensschema der restlichen Optionen von TUD-Script zu entsprechen.

<code>headfoot=&lt;Einstellung&gt;</code>	(siehe KOMA-Script-Optionen <code>headinclude</code> und <code>footinclude</code> )
---	---

Diese Option war für TUD-Script in der *Version v1.0* notwendig, um die parallele Verwendung der beiden Pakete **typearea** und **geometry** zu ermöglichen. Die Erstellung des Satzspiegels wurde komplett überarbeitet. Mittlerweile werden an das Paket **geometry** die Einstellungen für die KOMA-Script-Optionen `headinclude` und `footinclude` direkt weitergereicht, sodass die Option `headfoot` nicht mehr notwendig ist und deshalb entfernt wurde.

<code>partclear=&lt;Ein-Aus-Wert&gt;</code>	(siehe <code>cleardoublespecialpage</code> auf Seite 41)
<code>chapterclear=&lt;Ein-Aus-Wert&gt;</code>	(siehe <code>cleardoublespecialpage</code> auf Seite 41)

Beide Optionen sind in der neuen Option `cleardoublespecialpage` aufgegangen, womit ein konsistentes Layout erreicht wird. Die ursprünglichen Optionen entfallen.

<code>abstracttoc=&lt;Ein-Aus-Wert&gt;</code>	(siehe <code>abstract</code> auf Seite 42)
<code>abstractdouble=&lt;Ein-Aus-Wert&gt;</code>	(siehe <code>abstract</code> auf Seite 42)

Beide Optionen wurden in die Option `abstract` integriert und sind deshalb überflüssig.

<code>\logfile{&lt;Dateiname&gt;}</code>	(siehe <code>\headlogo</code> auf Seite 30)
--	---

Der Befehl `\logfile` wurde in `\headlogo` umbenannt, wobei die Funktionalität weiterhin bestehen bleibt.

`bookmarks=<Ein-Aus-Wert>` (siehe `tudbookmarks` auf Seite 50)

Die Option wurde umbenannt, um Überschneidungen mit `hyperref` zu vermeiden.

`\signatureheight` (Länge) (entfällt)

Die Höhe für die Zeile der Unterschriften wurde dehnbar gestaltet, eine etwaige Anpassung durch den Anwender ist nicht vonnöten.

`\titlecolldelim` (Bezeichner) (siehe `\titledelimiter` auf Seite 39)

Das Trennzeichen für Bezeichnungen beziehungsweise beschreibende Texte und dem eigentlichen Feld auf der Titelseite ist nicht mehr sprachabhängig und wurde umbenannt.

`\confirmationandrestriction` (siehe `\declaration` auf Seite 50)

`\restrictionandconfirmation` (siehe `\declaration` auf Seite 50)

Die beiden Befehle entfallen, stattdessen sollte entweder der Befehl `\declaration` oder die Umgebung `declarations` zusammen mit den Befehlen `\confirmation` und `\blocking` verwendet werden, wobei sich diese in der Umgebung in beliebiger Reihenfolge anordnen lassen.

`\location{<Ort>}` (siehe `\place` auf Seite 50)

In Anlehnung an andere  $\text{\LaTeX}$ -Pakete und -Klassen wurde `\location` in `\place` umbenannt.

## Aufgabenstellung

Die Umgebung für die Erstellung einer Aufgabenstellung für eine wissenschaftliche Arbeit wurde in das Paket `tudscrsvsupervisor` ausgelagert. Dieses muss für die Verwendung der Umgebung `task` und der daraus abgeleiteten standardisierten Form zwingend geladen werden.

`cdtask=<Einstellung>` (siehe `task` auf Seite 64)

`taskcompact=<Ein-Aus-Wert>` (entfällt)

`\taskcolwidth` (Länge) (entfällt)

Die Klassenoption `cdtask` ist komplett entfernt worden, alle Einstellungen, erfolgen direkt über das optionale Argument der Umgebung `task`. Die Variante eines kompakten Kopfes mit der Option `taskcompact` wird nicht mehr bereitgestellt. Die Möglichkeit zur manuellen Festlegung der Spaltenbreite für den Kopf der Aufgabenstellung mit `\taskcolwidth` wurde aufgrund der verbesserten automatischen Berechnung entfernt.

`\tasks{<Ziele>}{<Schwerpunkte>}` (siehe `\taskform` auf Seite 64)

`\focustext` (Bezeichner) (siehe `\focusname` auf Seite 68)

`\objectivestext` (Bezeichner) (siehe `\objectivesname` auf Seite 68)

Der Befehl `\tasks` wurde in `\taskform` umbenannt und in der Funktionalität erweitert. Die darin verwendeten Bezeichner wurden ebenfalls leicht abgewandelt.

`\studentid{<Matrikelnummer>}` (siehe `\matriculationnumber` auf Seite 35)

`\enrolmentyear{<Immatrikulationsjahr>}` (siehe `\matriculationyear` auf Seite 35)

`\submissiondate{<Datum>}` (siehe `\date` auf Seite 36)

`\birthday{<Geburtsdatum>}` (siehe `\dateofbirth` auf Seite 35)

`\birthplace{<Geburtsort>}` (siehe `\placeofbirth` auf Seite 35)

`\startdate{<Ausgabedatum>}` (siehe `\issuedate` auf Seite 65)

Alle Befehle wurden umbenannt und sind jetzt neben der Aufgabenstellung auch für die Titelseite im Corporate Design nutzbar.

<code>\studentidname</code> (Bezeichner)	(siehe <code>\matriculationnumbername</code> auf Seite 53)
<code>\enrolmentname</code> (Bezeichner)	(siehe <code>\matriculationyearname</code> auf Seite 53)
<code>\submissiontext</code> (Bezeichner)	(siehe <code>\datetext</code> auf Seite 52)
<code>\birthdaytext</code> (Bezeichner)	(siehe <code>\dateofbirthtext</code> auf Seite 53)
<code>\birthplacetext</code> (Bezeichner)	(siehe <code>\placeofbirthtext</code> auf Seite 53)
<code>\supervisorIiname</code> (Bezeichner)	(siehe <code>\supervisorothername</code> auf Seite 51)
<code>\defensetext</code> (Bezeichner)	(siehe <code>\defensedatetext</code> auf Seite 52)
<code>\starttext</code> (Bezeichner)	(siehe <code>\issuedatetext</code> auf Seite 68)
<code>\duetext</code> (Bezeichner)	(siehe <code>\duedatetext</code> auf Seite 68)

Die Bezeichner wurden in Anlehnung an die dazugehörigen Befehlsnamen umbenannt.

## B.1.2 Änderungen für TUD-Script v2.02

<code>\chapterheadingsvskip</code> (Länge)	(siehe <code>pageheadingsvskip</code> auf Seite 26)
--	---

Die vertikale Positionierung von Überschriften wurde aufgeteilt. Zum einen kann diese für Titel-, Teile- und Kapitelseiten (`chapterpage=true`) über die Option `pageheadingsvskip` geändert werden. Für Kapitelüberschriften (`chapterpage=false`) sowie den Titelkopf (`titlepage=false`) kann dies unabhängig davon mit `headingsvskip` erfolgen.

<code>\degree[&lt;Abk.&gt;]{&lt;Grad&gt;}</code>	(siehe <code>\graduation</code> auf Seite 38)
<code>\degreertext</code> (Bezeichner)	(siehe <code>\graduationtext</code> auf Seite 52)

Der Befehl wurde zur Erhöhung der Kompatibilität mit anderen Paketen umbenannt, der dazugehörige Bezeichner dahingehend angepasst.

<code>\restriction[&lt;Firma&gt;    &lt;Parameterliste&gt;]</code>	(siehe <code>\blocking</code> auf Seite 49)
<code>\restrictionname</code> (Bezeichner)	(siehe <code>\blockingname</code> auf Seite 53)
<code>\restrictiontext</code> (Bezeichner)	(siehe <code>\blockingtext</code> auf Seite 54)

Der Befehl wurde zur Erhöhung der Kompatibilität mit anderen Paketen umbenannt, die dazugehörigen Bezeichner dahingehend angepasst.

<code>\begin{tudpage}[&lt;Sprache&gt;    &lt;Parameterliste&gt;]</code>	
...	
<code>\end{tudpage}</code>	
<code>head=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>pagestyle</code> auf Seite 32)
<code>foot=&lt;Einstellung&gt;</code> (Parameter)	(siehe <code>pagestyle</code> auf Seite 32)

Diese beiden Parameter der Umgebung `tudpage` wurden in ihrer Funktionalität durch den Parameter `pagestyle` ersetzt.

## Änderungen im Paket tudscrsupervisor

Im Paket `tudscrsupervisor` gab es ein paar kleinere Anpassungen.

<code>\branch{&lt;Studienrichtung&gt;}</code>	(siehe <code>\discipline</code> auf Seite 35)
<code>\branchname</code> (Bezeichner)	(siehe <code>\disciplinename</code> auf Seite 53)

Für die Aufgabenstellung wurden der Befehl sowie der dazugehörige Bezeichner umbenannt.

<code>\contact{&lt;Kontaktperson(en)&gt;}</code>	(siehe <code>\contactperson</code> auf Seite 67)
<code>\contactname</code> (Bezeichner)	(siehe <code>\contactpersonname</code> auf Seite 69)
<code>\phone{&lt;Telefonnummer&gt;}</code>	(siehe <code>\telephone</code> auf Seite 67)
<code>\email{&lt;E-Mail-Adresse&gt;}</code>	(siehe <code>\emailaddress</code> auf Seite 35)

Alle genannten Befehle und Bezeichner wurden für den Aushang umbenannt.

### B.1.3 Änderungen für TUD-Script v2.03

`geometry=<Ein-Aus-Wert>` (siehe `cdgeometry` auf Seite 22)

Die Option `geometry` wurde zur Konsistenz sowie dem Vermeiden eines möglichen Konfliktes mit einer späteren KOMA-Script-Version umbenannt. Die Funktionalität bleibt bestehen.

`cdfonts=<Ein-Aus-Wert>` (siehe `cdfont` auf Seite 16)  
`din=<Ein-Aus-Wert>` (siehe `cdfont` auf Seite 16)

Die Option `cdfont` wurde erweitert und fungiert als zentrale Schnittstelle zur Schrifteinstellung.

`sansmath=<Ein-Aus-Wert>` (siehe `cdmath` auf Seite 19)

Die Option `sansmath` wurde aus Gründen der Konsistenz umbenannt. Zusätzlich wurde die Funktionalität erweitert.

`barfont=<Einstellung>` (siehe `cdhead` auf Seite 27)  
`widehead=<Ein-Aus-Wert>` (siehe `cdhead` auf Seite 27)

Die Optionen `barfont` und `widehead` wurden in der Option `cdhead` zusammengefasst.

```
\begin{tudpage}[<Sprache> || <Parameterliste>]
. . .
\end{tudpage}
color=<Farbe> (Parameter) (entfällt)
```

Der Parameter `color=<Farbe>` der `tudpage`-Umgebung wurde ersatzlos entfernt.

### B.1.4 Änderungen für TUD-Script v2.04

`fontspec=<Ein-Aus-Wert>` (entfällt)

Anstatt die Option `fontspec` zu aktivieren, kann einfach das Paket `fontspec` in der Dokumentpräambel geladen werden. Dadurch können anschließend zusätzliche Pakete genutzt werden, welche auf die Verwendung von `fontspec` angewiesen sind. Sollte die Option `fontspec` dennoch genutzt werden, müssen alle auf das Paket `fontspec` aufbauenden Einstellungen durch den Anwender mit `\AfterPackage{fontspec}{. . .}` verzögert werden. In Unterkapitel 2.1 auf Seite 15 sind weitere Hinweise zur Verwendung des Paketes `fontspec` zu finden.

### B.1.5 Änderungen für TUD-Script v2.05

`\pageheadingsvskip` (Länge) (siehe `pageheadingsvskip` auf Seite 26)  
`\headingsvskip` (Länge) (siehe `headingsvskip` auf Seite 26)

Die vertikale Positionierung von speziellen Überschriften erfolgt nicht mehr über die Längen `\headingsvskip` und `\pageheadingsvskip` sondern über die Optionen `headingsvskip` sowie `pageheadingsvskip`.

`\footlogoheight` (Länge) (siehe `footlogoheight` auf Seite 30)

Auch die Höhe der Logos im Fußbereich der `tudheadings`-Seitenstile wird von nun an mit der Option `footlogoheight` und nicht mehr mit der Länge `\footlogoheight` festgelegt.

## B.1.6 Änderungen für TUD-Script v2.06

v2.06

`cdoldfont` Voreinstellung: false

Achtung!

Mit der Version v2.06 wird standardmäßig **Open Sans** als Hausschrift verwendet. Um jedoch weiterhin ältere Dokumente mit den Schriften **Univers** und **DIN Bold** erzeugen zu können, wird diese Option bereitgestellt. *Diese kann ausschließlich als Klassenoption – oder für die Pakete **tudscrfonts** und **fix-tudscrfonts** als Paketoption – genutzt werden.* Eine späte Optionenwahl mit `\TUDoption` oder `\TUDoptions` ist nicht möglich. Wurden mit `cdoldfont=true` die alten Schriftfamilien aktiviert, kann jedoch weiterhin die Option `cdfont` genutzt werden.

`cdoldfont=false`

Das Verhalten ist äquivalent zu `cdfont=false`, die Hausschrift ist nicht aktiv.

`cdoldfont=true`

Es werden die alten Hausschriften **Univers** für den Fließtext sowie **DIN Bold** für Überschriften der obersten Gliederungsebenen bis einschließlich `\subsubsection` genutzt. Die Schriftstärke lässt sich mit `cdfont=true` respektive `cdfont=heavy` anpassen.

Für die TUD-Script-Klassen sowie die vom Paket **fix-tudscrfonts** unterstützten Dokumentklassen kann die für die Gliederungsebenen verwendete Schriftart angepasst werden.

`cdoldfont=nodin`

Für Überschriften wird **Univers** anstatt **DIN Bold** verwendet.

`cdoldfont=din`

Mit dieser Einstellung wird die Schrift **DIN Bold** in den Überschriften verwendet.

`cdoldfont=onlydin`

Hiermit werden nur die Überschriften in **DIN Bold** gesetzt, für den Fließtext kommt nicht **Univers** sondern die L<sup>A</sup>T<sub>E</sub>X-Standardschriften respektive die eines geladenen Schriftpaketes zum Einsatz.

<code>\univln</code>	<i>(entfällt)</i>
<code>\textuln{&lt;Text&gt;}</code>	<i>(entfällt)</i>
<code>\univrn</code>	<i>(entfällt)</i>
<code>\texturn{&lt;Text&gt;}</code>	<i>(entfällt)</i>
<code>\univbn</code>	<i>(entfällt)</i>
<code>\textubn{&lt;Text&gt;}</code>	<i>(entfällt)</i>
<code>\univxn</code>	<i>(entfällt)</i>
<code>\textuxn{&lt;Text&gt;}</code>	<i>(entfällt)</i>
<code>\univls</code>	<i>(entfällt)</i>
<code>\textuls{&lt;Text&gt;}</code>	<i>(entfällt)</i>
<code>\univrs</code>	<i>(entfällt)</i>
<code>\texturs{&lt;Text&gt;}</code>	<i>(entfällt)</i>
<code>\univbs</code>	<i>(entfällt)</i>
<code>\textubs{&lt;Text&gt;}</code>	<i>(entfällt)</i>
<code>\univxs</code>	<i>(entfällt)</i>
<code>\textuxs{&lt;Text&gt;}</code>	<i>(entfällt)</i>
<code>\dinbn</code>	<i>(entfällt)</i>
<code>\textdbn{&lt;Text&gt;}</code>	<i>(entfällt)</i>

Wird die Option `cdoldfont` nicht aktiviert, werden auch die Befehle zur expliziten Auswahl eines Schriftschnittes nicht mehr bereitgestellt. Stattdessen können `\cdfont` oder `\textcd{<Text>}` genutzt werden, welche in Abschnitt 2.1.1 zu finden sind.

<code>cdfont=din</code>	(entfällt, <i>siehe</i> <code>cdoldfont=din</code> )
<code>cdfont=nodin</code>	(entfällt, <i>siehe</i> <code>cdoldfont=nodin</code> )

Die Einstellungen für Überschriften sind mit der Umstellung auf **Open Sans** nicht mehr notwendig. Für die Verwendung der alten Schriftfamilien **Univers** und **DIN Bold** muss die Option `cdoldfont` aktiviert werden.

<code>clearcolor=&lt;Ein-Aus-Wert&gt;</code>	( <i>siehe</i> <code>cleardoublespecialpage</code> auf Seite 41)
--	--

Die Option `clearcolor=<Ein-Aus-Wert>` wurde zur Vereinheitlichung der Benutzerschnittstelle in `cleardoublespecialpage=<Einstellung>` integriert.

### Auszeichnungen in Überschriften

Für alle Gliederungsebenen bis einschließlich `\subsubsection` werden die Überschriften in Großbuchstaben der Schrift **DIN Bold** gesetzt, wenn diese mit den entsprechenden Einstellungen (`cdoldfont=true/onlydin`) aktiviert wurde. Hierfür wird intern `\MakeTextUppercase` aus dem Paket `textcase` genutzt, welches zusammen mit den alten Schriftfamilien geladen wird. Sollen bestimmte Kleinbuchstaben erhalten bleiben, ist `\NoCaseChange` zu verwenden.

**Beispiel:** In einem Kapitel wird ein einzelnes Wort in Minuskeln geschrieben:

```
\chapter{Überschrift mit \NoCaseChange{kleinem} Wort}
```

Die Schrift **DIN Bold** durfte laut Corporate Design nur mit Majuskeln (Großbuchstaben) verwendet werden, weshalb das beschriebene Vorgehen lediglich im *Ausnahmefall* anzuwenden ist. Die manuelle Nutzung sollte mit `\MakeTextUppercase{\textdbn{<Text>}}` geschehen.

<code>\ifdin{&lt;Dann-Teil&gt;}{&lt;Sonst-Teil&gt;}</code>	(entfällt)
--	------------

Der Befehl `\ifdin` prüft, ob die Schriftfamilie **DIN Bold** aktiv ist und führt in diesem Fall `{<Dann-Teil>}` aus, andernfalls `{<Sonst-Teil>}`.

## B.2 Das Paket `tudscrcomp` – Umstieg von anderen Klassen

Achtung!

Sollten Sie `tudbook`, `tudletter`, `tudfax`, `tudhaus`, `tudform` oder auch `tudmathposter` beziehungsweise eine der TUD-Script-Klassen in der Version v1.0 nie genutzt haben, können Sie dieses Unterkapitel ohne Weiteres überspringen. Sämtliche hier vorgestellten Optionen und Befehle sind in der aktuellen Version von TUD-Script obsolet.

Zu Beginn der Entwicklung von TUD-Script diente die Klasse `tudbook` als grundlegende Basis zur Orientierung. Ziel war es, sämtliche Funktionalitäten dieser Klasse beizubehalten und zusätzlich den vollen Funktionsumfang der KOMA-Script-Klassen nutzbar zu machen. Bei der kompletten Neuimplementierung der TUD-Script-Klassen wurde sehr viel verändert und verbessert. Ein Teil der implementierten Optionen und Befehle war jedoch bereits in der Version v1.0 von TUD-Script unerwünschte Relikte, mit denen lediglich die Kompatibilität zur `tudbook`-Klasse und ihren Derivaten gewährleistet werden sollte. Mit der Version v2.00 wurden einige der unnötigen Befehle und Optionen aus Gründen der Konsistenz nur umbenannt, andere wiederum wurden vollständig entfernt oder über neue Befehle und Optionen in ihrer Funktionalität ersetzt und erweitert.

Das Paket `tudscrcomp` dient der Überführung von Dokumenten, welche entweder mit der `tudbook`-Klasse, ihren Derivaten, der Klasse `tudmathposter` oder mit TUD-Script in der Version v1.0 erstellt wurden, auf TUD-Script v2.06o. Falls Sie das Paket verwenden wollen, sollte

Achtung!



es **direkt** nach der Dokumentklasse geladen werden. Andernfalls kann es im Zusammenhang mit anderen Paketen zu Problemen kommen.

Es werden einige Optionen und Befehle bereitgestellt, welche von den zuvor genannten Klassen definiert werden, um das entsprechende Verhalten nachzuahmen. Damit soll vor allem die Kompatibilität bei einer Änderung der Dokumentklasse sichergestellt werden. Die Intention ist, Dokumente möglichst schnell und einfach auf die TUD-Script-Klassen portieren zu können. Des Weiteren ist beschrieben, wie sich die Funktionalität ohne eine Verwendung des Paketes **tudscrcomp** mit den Mitteln von TUD-Script umsetzen lassen. Für den Satz neuer Dokumente wird empfohlen, auf den Einsatz dieses Paketes komplett zu verzichten und stattdessen direkt die TUD-Script-Befehle zu nutzen.

<code>\einrichtung{&lt;Fakultät&gt;}</code>	(identisch zu <code>\faculty{&lt;Fakultät&gt;}</code> )
<code>\fachrichtung{&lt;Einrichtung&gt;}</code>	(identisch zu <code>\department{&lt;Einrichtung&gt;}</code> )
<code>\institut{&lt;Institut&gt;}</code>	(identisch zu <code>\institute{&lt;Institut&gt;}</code> )
<code>\professur{&lt;Lehrstuhl&gt;}</code>	(identisch zu <code>\chair{&lt;Lehrstuhl&gt;}</code> )

Dies sind die deutschsprachigen Befehle für den Kopf im Corporate Design.

<code>serifmath</code>	(identisch zu <code>cdmath=false</code> )
------------------------	---

Die Funktionalität wird durch die Option `cdmath` bereitgestellt.

<code>\tudfont{&lt;Schriftart&gt;}</code>	(identisch zu <code>\cdfont{&lt;Schriftart&gt;}</code> )
---	--

Die direkte Auswahl der Schriftart sollte mit `\cdfont` erfolgen. Zusätzlich gibt es den Befehl `\textcd`, mit dem die Auszeichnung eines bestimmten Textes in einer anderen Schriftart erfolgen kann, ohne die Dokumentschrift umzuschalten.

## B.2.1 Optionen und Befehle aus **tudbook & Co.**

Die nachfolgenden Optionen, Umgebungen sowie Befehle werden – zumindest teilweise – von den Klassen **tudbook**, **tudletter**, **tudfax**, **tudhaus**, **tudform** sowie dem Paket **tudthesis** und TUD-Script in der *Version v1.0* definiert. Diese werden durch das Paket **tudscrcomp** für TUD-Script v2.060 zur Verfügung gestellt.

<code>colortitle</code>	(identisch zu <code>cdtitle=color</code> )
<code>nocolortitle</code>	(identisch zu <code>cdtitle=true</code> )

Die Funktionalität wird durch die Option `cdtitle` bereitgestellt.

<code>\moreauthor{&lt;Autorenzusatz&gt;}</code>	(identisch zu <code>\authormore{&lt;Autorenzusatz&gt;}</code> )
---	---

Ursprünglich war diese Befehl für das Unterbringen aller möglichen, zusätzlichen Autoreninformationen gedacht. Auch der Befehl `\authormore` ist ein Rudiment davon. Empfohlen wird stattdessen die Verwendung der Befehle `\dateofbirth`, `\placeofbirth`, `\matriculationnumber` und `\matriculationyear` sowie `\course` und `\discipline`.

<code>\submitdate{&lt;Datum&gt;}</code>	(identisch zu <code>\date{&lt;Datum&gt;}</code> )
---	---

Die Funktionalität wird durch den erweiterten Standardbefehl `\date` abgedeckt.

<code>\supervisorII{&lt;Name&gt;}</code>	(identisch zur Verwendung von <code>\and</code> innerhalb von <code>\supervisor</code> )
--	--

Es ist `\supervisor{<Name> \and <Name>}` statt `\supervisorII{<Name>}` zu verwenden.

<code>\supervisedby{&lt;Bezeichnung&gt;}</code>	(siehe <code>\supervisorname</code> )
<code>\supervisedIIby{&lt;Bezeichnung&gt;}</code>	(siehe <code>\supervisorothername</code> )
<code>\submittedon{&lt;Bezeichnung&gt;}</code>	(siehe <code>\datetext</code> )

Zur Änderung der Bezeichnung der Betreuer sollten die sprachabhängigen Bezeichner wie in Unterkapitel 2.4 beschrieben angepasst werden. Eine Verwendung der hier beschriebenen Befehle entfernt die Abhängigkeit der Bezeichner von der verwendeten Sprache.

`ddcfooter` (identisch zu `ddcfoot=true`)

Die Funktionalität wird durch die Option `ddcfoot` bereitgestellt.

`\dissertation`

Die Funktionalität kann durch die Befehle `\thesis{diss}` und `\referee` sowie die Bezeichner `\refereename` und `\refereeothername` dargestellt werden.

`\chapterpage`

Durch diesen Befehl können Kapitelseiten konträr zur eigentlichen Einstellung aktiviert oder deaktiviert werden. Prinzipiell ist dies auch durch eine Änderung der Option `chapterpage` möglich. Allerdings wird davon abgeraten, da dies zu einem inkonsistenten Layout innerhalb des Dokumentes führt.

```
\begin{theglossary}[<Präambel>]
. . .
\end{theglossary}
\glossitem{<Begriff>}
```

Die **tudbook**-Klasse stellt eine rudimentäre Umgebung für ein Glossar bereit. Allerdings gibt es dafür bereits zahlreiche und besser implementierte Pakete. Daher wird für diese Umgebung keine Portierung vorgenommen, sondern lediglich die ursprüngliche Definition übernommen. Allerdings sein an dieser Stelle auf wesentlich bessere Lösungen wie beispielsweise das Paket **glossaries** oder – mit Abstrichen – das nicht ganz so umfangreiche Paket **nomenc1** verwiesen.

## B.2.2 Optionen und Befehle aus **tudmathposter**

v2.05

Die Klasse **tudmathposter** wird – im Gegensatz zu den zuvor genannten Klassen von Klaus Bergmann – weiterhin gepflegt und kann bedenkenlos zum Setzen von Postern im A0-Format verwendet werden. Dennoch gab es vermehrt Anfragen bezüglich einer Posterklasse auf Basis der TUD-Script-Klassen, um beispielsweise die Schriftgröße oder auch das Papierformat einfach anpassen zu können. Um von **tudmathposter** einen möglichst einfachen Übergang auf **tudscrposter** zu gewährleisten, kann zusätzlich letzterer Klasse das Paket **tudscrcomp** geladen werden, welches die nachfolgend erläuterten Anwenderbefehle bereitstellt. Von **tudscrcomp** werden außerdem einige Pakete geladen, welche normalerweise durch **tudmathposter** eingebunden werden. Dies sind **amsmath**, **amsfonts**, **calc**, **textcomp** sowie **tabularx**.

Es ist nicht beabsichtigt, dass bei einem Umstieg von **tudmathposter** auf **tudscrposter** in Verbindung mit **tudscrcomp** das Ausgabeergebnis identisch ist. Vielmehr soll damit die Möglichkeit geschaffen werden, auf **tudmathposter** basierende Dokumente auf die **tudscrposter** zu überführen. In jedem Fall sollte bei einem Umstieg auf TUD-Script-Posterklasse beachtet werden, dass für diese Klasse eine explizite Wahl der Schriftgröße über die Option `fontsize` notwendig ist. Um kongruent zur Klasse **tudmathposter** zu bleiben, ist die Wahl einer Schriftgröße von `fontsize=34. . .36pt` sinnvoll. Für weitere Informationen zu diesem Thema sind

in Abschnitt 3.1.1 vorhanden. Weiterhin sollte für ein ähnliches Ausgabeergebnis die Absatzformatierung über die KOMA-Script-Option `parskip=half-` eingestellt werden. Ein blaues DRESDEN-concept-Logo im Fußbereich lässt sich über `ddcfoot=blue` aktivieren.

<code>tudmathposterfoot</code>
--------------------------------

Durch die Klasse **tudmathposter** wird der Fußbereich zweispaltig jedoch asymmetrisch und ohne Überschriften innerhalb der beiden Spalten gesetzt. Dieses Verhalten lässt sich mit der Option `tudmathposterfoot` auswählen. Alternativ kann auch `cdfoot=tudmathposter` respektive `cdfoot=tudscrposter` zum Aktivieren beziehungsweise Deaktivieren verwendet werden.

<code>bluebg</code>	(identisch zu <code>backcolor=true</code> )
---------------------	---

Mit der Klasse **tudscrposter** lässt sich das Verhalten mit der Option `backcolor` umsetzen.

<code>\telefon{&lt;Telefonnummer&gt;}</code>	(identisch zu <code>\telephone{&lt;Telefonnummer&gt;}</code> )
<code>\fax{&lt;Telefaxnummer&gt;}</code>	(identisch zu <code>\telefax{&lt;Telefaxnummer&gt;}</code> )
<code>\email{&lt;E-Mail-Adresse&gt;}</code>	(identisch zu <code>\emailaddress*{&lt;E-Mail-Adresse&gt;}</code> )
<code>\homepage{&lt;URL&gt;}</code>	(identisch zu <code>\webpage*{&lt;URL&gt;}</code> )

Dies sind die von **tudmathposter** definierten Befehle für die Felder im vordefinierten Fußbereich des Posters. Es ist dabei insbesondere zu beachten, dass die angegebene E-Mail-Adresse sowie die URL nicht automatisch formatiert werden.

<code>\fusszeile{&lt;Inhalt&gt;}</code>	(identisch zu <code>\footcontent{&lt;Inhalt&gt;}</code> )
<code>\footcolumn0{&lt;Inhalt&gt;}</code>	(identisch zu <code>\footcontent{&lt;Inhalt&gt;}</code> )
<code>\footcolumn1{&lt;Inhalt&gt;}</code>	(identisch zu <code>\footcontent{&lt;Inhalt&gt;[*]}</code> )
<code>\footcolumn2{&lt;Inhalt&gt;}</code>	(identisch zu <code>\footcontent*{&lt;Inhalt&gt;}</code> )

Mit diesen Befehlen kann die Gestalt des Fußes angepasst werden, wobei entweder der Bereich über die gesamte Breite (`\fusszeile`, `\footcolumn0`) oder lediglich die linke (`\footcolumn1`) respektive die rechte Spalte (`\footcolumn2`) angepasst wird. Für zusätzliche Hinweise zur Anpassung des Fußbereichs – insbesondere für die Schriftformatierung – sollte die Beschreibung von `\footcontent` in Abschnitt 2.2.3 auf Seite 30 zu Rate gezogen werden.

<code>\zweitlogo{&lt;Dateiname&gt;}</code>	(identisch zu <code>\headlogo{&lt;Dateiname&gt;}</code> )
<code>\institutslogo{&lt;Dateiname&gt;}</code>	(siehe <code>\footlogo</code> )
<code>\drittlogo{&lt;Dateiname&gt;}</code>	(siehe <code>ddc</code> und <code>ddcfoot</code> )

Für die Angabe von Logos für den Kopf- und Fußbereich existieren diese Befehle. Bei der Verwendung von `\institutslogo{<Dateiname>}` ist zu beachten, dass die angegebene Datei sehr weit rechts im Fußbereich des Posters gesetzt wird. Dabei kommt bei der Verwendung im Hintergrund der von TUD-Script für das Setzen von Logos im Fußbereich tatsächlich vorgesehene Befehl in der Form `\footlogo{, , , , , <Dateiname> , }` zum Einsatz. Das Makro `\drittlogo` wird von **tudmathposter** für die Angabe eines DRESDEN-concept-Logos im rechten Seitenfuß bereitgestellt. Für die TUD-Script-Klassen gibt es hierfür die Optionen `ddc` beziehungsweise `ddcfoot`.

<code>\zweitlogo{&lt;Definition&gt;}</code>	(keine Funktionalität, siehe <code>\headlogo</code> )
<code>\institutslogo{&lt;Definition&gt;}</code>	(keine Funktionalität, siehe <code>\footlogo</code> )
<code>\drittlogo{&lt;Definition&gt;}</code>	(keine Funktionalität, siehe <code>ddc</code> und <code>ddcfoot</code> )

Mit diesen Befehlen kann der Anwender bei **tudmathposter** die Definition für das Einbinden diverser Logos selber vornehmen. Dies ist für TUD-Script nicht vorgesehen. Im Zweifel sollten die Möglichkeiten der korrelierenden Befehle genutzt werden. Bei der Verwendung eines dieser Makros wird lediglich eine Warnung ausgegeben.

```
\topsection[<Kurzform>]{<Überschrift>}
\topsubsection[<Kurzform>]{<Überschrift>}
```

Der Grund für die Existenz dieser beiden Befehle bei **tudmathposter** ist nicht ohne Weiteres nachvollziehbar. Beide entsprechen in ihrem Verhalten den Standardbefehlen `\section` und `\subsection`, setzen allerdings keinen vertikalen Abstand vor der erzeugten Überschrift. Auch wenn das aus typographischer Sicht wohl eher unvorteilhaft ist, werden diese beiden Befehle bereitgestellt.

```
\centersection[<Kurzform>]{<Überschrift>}
\centersubsection[<Kurzform>]{<Überschrift>}
\topcentersection[<Kurzform>]{<Überschrift>}
\topcentersubsection[<Kurzform>]{<Überschrift>}
```

Weiterhin werden auch noch eigene Makros zum Setzen zentrierter Überschriften definiert – ein simples Umdefinieren von `\raggedsection` ist dafür im Normalfall absolut ausreichend. Und um die Sache vollständig zu machen, gibt es die zentrierten Überschriften auch noch ohne vorgelagerten, vertikalen Abstand.

```
\begin{farbtabellen}
. . .
\end{farbtabellen}
\blautabelle
\grautabelle
```

Wird innerhalb der `farbtabellen`-Umgebung eine Tabelle gesetzt, so werden die Zeilen alternierend farbig hervorgehoben. Standardmäßig sind hierfür leichte Blautönen eingestellt, was auch jederzeit mit dem Aufruf von `\blautabelle` wiederhergestellt werden kann. Alternativ zu dieser Darstellung kann mit `\grautabelle` auf eine Verwendung von leichten Grautönen umgestellt werden.

```
\begin{figurehere}
. . .
\end{figurehere}
\begin{tablehere}
. . .
\end{tablehere}
```

Diese beiden Umgebungen können genutzt werden, um nicht-gleitende Abbildungen oder Tabellen zu setzen.

```
\schnitttrand
```

Wird `\schnitttrand` vor dem Laden des Paketes **tudscrcomp** definiert, so wird der Inhalt des Befehls als Längenwert interpretiert. Dieser wird verwendet, um den zuvor festgelegten Satzspiegel über die drei Parameter `paper=<Papierformat>`, `layout=<Zielformat>` und `layoutoffset=<Längenwert>` des Befehls `\geometry` aus dem Paket **geometry** zu setzen und das erzeugte Papierformat um den gegebenen Längenwert an allen Rändern zu vergrößern. Somit wird eine Beschnittzugabe hinzugefügt, *ohne dabei die Seitenränder des Entwurfslayouts anzupassen*. In Unterkapitel 9.21 auf Seite 124 sind zusätzliche Informationen zu diesem Thema zu finden.

Anhang C

## Identifikation von TUD-Script

Im TUD-Script-Bundle gibt es neben den Klassen selbst auch noch zusätzliche Pakete. Ein Teil dieser Pakete – genauer **tudscrsupervisor** und **tudscrcomp** – sind ausschließlich mit den TUD-Script-Klassen nutzbar, andere wiederum – die beiden Pakete für Belange des Corporate Designs **tudscrfonts** (Schriften) und **tudscrcolor** (Farben) sowie die davon vollkommen unabhängigen Pakete **mathswap** und **twocolfix** – können mit allen existierenden  $\LaTeX 2_{\epsilon}$ -Dokumentklassen genutzt werden. Sämtliche Klassen und Pakete aus dem TUD-Script-Bundle enthalten Befehle, welche diese als Bestandteil identifizieren.

v2.04

`\TUDScript`

Diese Anweisung setzt das Logo respektive die Wortmarke „TUD-Script“ in serifenloser Schrift und mit leichter Sperrung des in Versalien gesetzten Teils. Dieser Befehl wird von allen Klassen und Paketen des TUD-Script-Bundles mit `\DeclareRobustCommand`.

v2.04

`\TUDScriptClassName`

Die Bezeichnung der jeweiligen, im Dokument verwendeten TUD-Script-Klasse ist im Makro `\TUDScriptClassName` abgelegt. Soll also in Erfahrung gebracht werden, ob – und wenn ja, welche – TUD-Script-Klasse verwendet wird, so kann einfach auf diese Anweisung getestet werden. KOMA-Script stellt zusätzlich noch die beiden Anweisungen `\KOMAClassName` und `\ClassName` bereit, welche den Namen der zugrundeliegenden KOMA-Script-Klasse sowie die durch diese ersetzte Standardklasse enthalten.

v2.04

`\TUDScriptVersion`

v2.05

`\TUDScriptVersionNumber`

In `\TUDScriptVersion` ist die Hauptversion von TUD-Script in der Form

*<Datum> <Version> TUD-Script*

abgelegt. Die Version ist für alle Klassen und Pakete des TUD-Script-Bundles gleich und kann nach dem Laden einer Klasse oder eines Paketes abgefragt werden. Beispielsweise wurde diese Anleitung mit „2022/08/09 v2.06o TUD-Script“ erstellt.

Eventuell will der Anwender auf die aktuell verwendete Version von TUD-Script prüfen, um gegebenenfalls eigene Anpassungen in Abhängigkeit der verwendeten Version vorzunehmen. Hierfür kann `\TUDScriptVersionNumber` verwendet werden. Darin ist alleinig die Versionsnummer enthalten. Die für das Handbuch verwendete Version lautet „2.06o“.

# Index

Die Formatierung der Einträge in allen aufgeführten Indexen ist folgendermaßen aufzufassen: **Zahlen in fetter Schrift** verweisen auf die **Erklärung** zu einem Stichwort, wobei in der digitalen Fassung dieses Handbuchs dieser Eintrag selbst ein Hyperlink zu seiner Erläuterung ist. Seitenzahlen in normaler Schriftstärke hingegen deuten auf zusätzliche Informationen, wobei diese für *kursiv hervorgehobene Zahlen* als besonders *wichtig* erachtet werden.

Bei Einträgen für Klassen- und Paketoptionen, Umgebungen und Befehlen sowie Seitenstilen, Schriftelementen und Farben respektive Längen und Zählern, zu denen keine direkte **Erklärung** existiert – und diese dementsprechend auch nicht als Hyperlink dargestellt sind – handelt es sich um zusätzliche Hinweise für Elemente, die nicht durch TUD-Script sondern von anderen Klassen oder Paketen zur Verfügung gestellt werden, welche im Suffix des entsprechenden Eintrages angegeben werden.

## Allgemeiner Index

Die im Folgenden aufgelisteten Schlagworte sollen für den Umgang mit  $\text{\LaTeX} 2_{\epsilon}$  im Allgemeinen sowie dem TUD-Script-Bundle im Speziellen sowohl Antworten bei generellen Fragen liefern als auch Lösungen für typische Probleme bereitstellen. Falls ein gesuchter Begriff hier nicht auftaucht oder trotz vorhandener Hinweise kein zufriedenstellendes Ergebnis erzielt werden kann, sollte das TUD- $\text{\LaTeX}$ -Forum erster Anlaufpunkt sein, um weitere Hilfe bei der Nutzung von TUD-Script zu erhalten.

### A

Abbildungen ..... *siehe* Grafiken  
Abkürzungsverzeichnis ..... 107  
Absatzauszeichnung ..... 120  
Abschlussarbeit **37, 52**, *siehe auch* Typisierung  
Änderungen → Änderungsliste ..... 166  
Aktualisierung ..... *siehe* Update  
Aufgabenstellung ..... **64, 65**, 50  
Aufzählungen ..... *siehe* Listen  
Aushang ..... **66, 67**, 50  
Autorenangaben ..... 35, 53, *siehe auch* Titel

### B

Befehle → Index der Befehle ..... 157  
Befehlsdeklaration ..... 101, 104, 112  
    Geltungsbereich ..... 118  
Berechnungen ..... 112  
Beschnittzugabe ..... **124, 60**, 112  
Betreuer ..... 38, 51  
Bezeichner ..... **50–54, 67–69**, 38, 102  
    → Index der Bezeichner ..... 160  
*biber* (Anwendungssoftware) ..... 106, 116  
Bindekorrektur ..... *siehe* Satzspiegel

### C

Changelog → Änderungsliste ..... 166  
*Citavi* (Anwendungssoftware) ..... 116  
Cover ..... *siehe* Umschlagseite

### D

Dateien → Index der Dateien etc. .... 162  
Datum ..... 36, 103, 27, 48, 52  
    Abgabedatum ..... 36, 52  
    Geburtsdatum ..... 35, 53  
    Verteidigungsdatum ..... 36, 52  
Deinstallation ..... 131  
*DeLig* (Anwendungssoftware) ..... 104  
Dezimaltrennzeichen *siehe* Zifferngruppierung  
Distribution ..... 10

*MacTeX* ..... *siehe* *MacTeX* (Distribution)

*MiKTeX* ..... *siehe* *MiKTeX* (Distribution)

*TeX Live* ..... *siehe* *TeX Live* (Distribution)

doppelseitiger Satz ..... *siehe* Satzspiegel

DRESDEN-concept-Logo ..... *siehe* Layout

Drittlogo ..... *siehe* Layout

### E

*Eclipse* (Anwendungssoftware) ..... 116

Eingabekodierung ..... 113

Einheiten ..... 106, 122

### F

Fachreferent ..... *siehe* Referent

Farben ... 71–73, 100, 112, *siehe auch* Layout

    → Index der Farben ..... 161

    Farbmodell ..... 73

Felder ..... **42**

Formelzeichenverzeichnis ..... 107

Fußnoten ..... 103, 120

Fußzeile ..... *siehe* Layout

### G

Gleitobjekte ..... 107, 108

    Platzierung ..... 125

Gliederung ..... *siehe* Layout | Überschriften

Glossar ..... 107

*GNU Aspell* (Anwendungssoftware) ..... 104

Grafiken ..... 109, 110, 100, 123, *siehe auch*

    Gleitobjekte

Griechische Buchstaben ..... 19, 20

Gutachten ..... **65, 66**, 50

Gutachter ..... 38, 51

### H

Hauptklassen ..... **15**

Hochschullehrer ..... 51

*Hunspell* (Anwendungssoftware) ..... 104



- I
- Inhaltsverzeichnis ..... 50, 120
  - Inkscape** (Anwendungssoftware) ..... 110, 123
  - Installation ..... *siehe* Update
    - Nutzerinstallation ..... 132
    - Schriftinstallation ..... **133–137**
- J
- JabRef** (Anwendungssoftware) ..... 116
- K
- Kapitelseiten ..... *siehe* Layout
  - Kapitelüberschriften ..... *siehe* Layout
  - Kile** (Anwendungssoftware) ..... 116
  - Klassen → Index der Dateien etc. .... 162
  - Klassenoptionen ..... *siehe* Optionen
  - Kollaboratives Schreiben ..... 36, 35
  - Kolumnentitel ..... *siehe* Layout
  - Kompatibilität ..... **55, 100, 139**
    - tudbook** ..... 144–146
    - tudmathposter** ..... 144–148
  - Kopfzeile ..... *siehe* Layout
  - Kurzfassung ..... *siehe* Zusammenfassung
- L
- Längen → Index der Längen ..... 162
  - LanguageTool** (Anwendungssoftware) ..... 104
  - L<sup>A</sup>T<sub>E</sub>X** (Textsatzsystem) ..... 123
  - LaTeXDraw** (Anwendungssoftware) ..... 110
  - Layout ..... 22–42
    - DRESDEN-concept-Logo ..... 30, 31
    - Drittlogo ..... 30
    - Farben ..... 71–73, 24–26, 42
    - Fußzeile ..... 24, 29–32
    - Kapitelseiten ..... 26, 40, 41
    - Kolumnentitel ..... 29
    - Kopfzeile ..... 24, 26, 27, 30–32
    - Querbalken ..... 26, 27
    - Rückseiten ..... 41
    - Seitenränder ..... *siehe* Satzspiegel
    - Seitenstil ..... 26, 27, 22, 24–32
    - Teileseiten ..... 26, 40, 41
    - Titel ..... *siehe* Titel
    - Überschriften ..... 24–26, 40, 41
    - Umschlagseite ..... *siehe* Umschlagseite
    - Zweitlogo ..... 30, 31
  - LEd** (Anwendungssoftware) ..... 116
  - Leerraum ..... 21, 55, *siehe auch*
    - Absatzauszeichnung
  - Leerseiten ..... *siehe* Vakantseiten
  - Lesezeichen ..... 110, 111, 50, 53
  - Ligaturen ..... 104
  - Listen ..... 106
  - Literaturverzeichnis ..... 106, 119
  - Lokalisierung ..... *siehe* Bezeichner
  - LuaL<sup>A</sup>T<sub>E</sub>X** (Textsatzsystem) . 16, 101, 102, 104, 113, 117
  - LyX** (Anwendungssoftware) ..... 116
- M
- MacT<sub>E</sub>X** (Distribution) ..... 10, 11, 134, 135
  - Makros ..... *siehe* Befehle
  - Mathematisatz 19–21, 75, 98, 105, 105, 106, *siehe auch* Einheiten; Griechische Buchstaben; Zifferngruppierung
  - Mehrspaltsatz ..... 103
  - MiK<sub>E</sub>T<sub>E</sub>X** (Distribution) ..... 10, 134, 135, 132
  - MiK<sub>E</sub>T<sub>E</sub>X Portable** (Distribution) ..... 136
  - Minimalbeispiel ..... 79–97, 112, 113
    - Abschlussarbeit ..... 87
    - Aufgabenstellung ..... 89
    - Aushang ..... 93
    - Dissertation ..... 84
    - Dokument ..... 79
    - Gutachten ..... 91
    - Kollaboratives Schreiben ..... 87, 89
    - Poster ..... 95
- N
- Nutzerinstallation ..... *siehe* Installation
- O
- OpenType-Schriften ..... 16
  - Optionen ..... **15**
    - Index der Optionen ..... 155
  - Outline-Eintrag ..... *siehe* Lesezeichen
- P
- Paketabhängigkeiten ..... 112
  - Pakete → Index der Dateien etc. .... 162
  - Papierformat ..... **59, 123**
  - Parameter ..... *siehe* Befehle
  - pdfL<sup>A</sup>T<sub>E</sub>X** (Textsatzsystem) 101, 102, 109, 110, 113, 117, 122, 123, 125
  - Posterklasse ..... **58**

- Professor ..... *siehe* Hochschullehrer
- Q**
- Quelltextdokumentation ..... 113
- Querbalken ..... *siehe* Layout
- Querverweise ..... 110, 111, 118
- R**
- Randnotizen ..... 111
- Rechtschreibung ..... 104
- Referent ..... 38, 51
- S**
- Satzspiegel ..... 22, 23, 55, 100
- Bindekorrektur ..... **23**
- doppelseitig ..... 22, 29, 33, 39–41
- mehrspaltig ..... 103
- zweispaltig ..... 75, 103, 33, 42, 44
- Schnittmarken ..... **124**, 60, 112
- Schriftart ..... **100**, 15, 16, 18, 104, 105
- Schriftauszeichnung ..... 18, 118
- Schriftelemente ..... **42**, 29, 35, 40, 41
- Index der Schriftelemente ..... 161
- Schriftgröße ..... **59**, 21, 55, 105
- Schriftstärke ..... 16, 18, 25, 100
- Seitenränder ..... *siehe* Satzspiegel
- Seitenstile ..... *siehe* Layout
- Index der Seitenstile ..... 161
- Selbstständigkeitserklärung ... **46–50**, 53, 54
- Silbentrennung ..... *siehe* Worttrennung
- Sonderzeichen ..... 105
- Sperrvermerk ..... **46–50**, 53, 54
- Sprachunterstützung ..... 102, *siehe auch*
- Bezeichner; Worttrennung
- Sprungmarken ..... *siehe* Lesezeichen
- Symbole ..... 121
- Symbolverzeichnis ..... 107
- T**
- Tabellen 108, 109, 119, *siehe auch* Gleitobjekte
- Tausendertrennzeichen ..... *siehe*
- Zifferngruppierung
- Teileseiten ..... *siehe* Layout
- Teileüberschriften ..... *siehe* Layout
- T<sub>E</sub>X Live** (Distribution) ..... 10, 11, 134, 135, 132
- T<sub>E</sub>X Live Portable** (Distribution) ..... 136
- T<sub>E</sub>Xlipse** (Anwendungssoftware) ..... 116
- T<sub>E</sub>Xmaker** (Anwendungssoftware) ..... 116
- T<sub>E</sub>XnicCenter** (Anwendungssoftware) ..... 116
- T<sub>E</sub>Xstudio** (Anwendungssoftware) ..... 116
- T<sub>E</sub>Xworks** (Anwendungssoftware) ..... 116
- Titel ..... **33–39**, 25, 32, 41, 50
- Felder ..... 35–39, 50–52
- Trennzeichen ..... 39
- Umschlagseite ..... *siehe* Umschlagseite
- Trennmuster ..... *siehe* Worttrennung
- Tutorials ..... **98**
- Type1-Schriften ..... 16
- Typisierung ..... 37, 38, 52
- Typografie ..... 101–104, 121
- U**
- Überfüllung ..... *siehe* Beschnittzugabe
- Umgebungen ..... *siehe* Befehle
- Umschlagseite ..... **34**, **35**, 25, 32, 50
- Update ..... **132**
- V**
- Vakatseiten ..... 40–42
- Vektorgrafiken ..... *siehe* Grafiken
- Versalien ..... *siehe* Schriftauszeichnung
- Version → Änderungsliste ..... 166
- Verzeichnisse ..... 106, 107
- W**
- WinEdt** (Anwendungssoftware) ..... 116
- Worttrennung ..... **102**, 103, 104
- X**
- X<sub>Y</sub>TEX** (Textsatzsystem) ..... 16, 101, 102, 117
- Z**
- Zähler
- Index der Zähler ..... 162
- Zeichensatzkodierung ..... 101
- Zeilenabstand ..... 102
- Ziffernform ..... 17
- Zifferngruppierung ..... 75, 98, 105, 106
- Zitate ..... 103
- Zusammenfassung ..... **42–46**, 53
- zweiseitiger Satz ..... *siehe* Satzspiegel
- zweispaltiger Satz ..... *siehe* Satzspiegel
- Zweitlogo ..... *siehe* Layout

## Klassen- und Paketooptionen

Dies ist der Index aller im Handbuch erläuterten sowie erwähnten Optionen, wobei bei den meisten auch Untereinträge mit Seitenangaben für spezielle Wertzuweisungen existieren.

<b>A</b>		<b>cd=true</b> .....	<b>60</b>
<b>abstract</b> .....	<b>43–46, 58</b>	<b>cdchapter</b> .....	<b>25</b>
<b>abstract=chapter</b> .....	<b>43, 44</b>	<b>cdcover</b> .....	<b>25, 34</b>
<b>abstract=false</b> .....	<b>43, 45</b>	<b>cdcover=false</b> .....	<b>34</b>
<b>abstract=fill</b> .....	<b>44, 45</b>	<b>cdcover=true</b> .....	<b>34</b>
<b>abstract=heading</b> .....	<b>43</b>	<b>cdfont</b> .....	<b>16–19, 25, 32–34, 58, 73</b>
<b>abstract=markboth</b> .....	<b>44, 43</b>	<b>cdfont=false</b> .....	<b>16, 19, 28</b>
<b>abstract=multiple</b> .....	<b>44, 45</b>	<b>cdfont=head</b> .....	<b>18</b>
<b>abstract=nofill</b> .....	<b>44, 45</b>	<b>cdfont=heavy</b> .....	<b>17</b>
<b>abstract=nomarkboth</b> .....	<b>44</b>	<b>cdfont=heavybold</b> .....	<b>17, 16</b>
<b>abstract=notoc</b> .....	<b>43</b>	<b>cdfont=heavyhead</b> .....	<b>18</b>
<b>abstract=section</b> .....	<b>43</b>	<b>cdfont=liningfigures</b> .....	<b>17</b>
<b>abstract=single</b> .....	<b>44, 45</b>	<b>cdfont=math</b> .....	<b>17</b>
<b>abstract=toc</b> .....	<b>43</b>	<b>cdfont=nohead</b> .....	<b>18</b>
<b>abstract=tocleveldown</b> .....	<b>43</b>	<b>cdfont=nomath</b> .....	<b>17</b>
<b>abstract=tocmultiple</b> .....	<b>43</b>	<b>cdfont=normalbold</b> .....	<b>17</b>
<b>abstract=true</b> .....	<b>43</b>	<b>cdfont=oldstylefigures</b> .....	<b>17</b>
<b>automark (sclayer-scrpage)</b> .....	<b>43, 47, 48</b>	<b>cdfont=slgreek</b> .....	<b>17</b>
		<b>cdfont=teggreek</b> .....	<b>17</b>
<b>B</b>		<b>cdfont=true</b> .....	<b>16, 17</b>
<b>backcolor (tudscrposter)</b> .....	<b>61</b>	<b>cdfont=upgreek</b> .....	<b>17</b>
<b>backcolor=false</b> .....	<b>60</b>	<b>cdfoot</b> .....	<b>27, 29, 31–34, 60, 61</b>
<b>backcolor=&lt;Farbe&gt;</b> .....	<b>60</b>	<b>cdfoot=color</b> .....	<b>29</b>
<b>backcolor=true</b> .....	<b>60</b>	<b>cdfoot=false</b> .....	<b>29</b>
<b>BCOR (typearea)</b> .....	<b>15, 23</b>	<b>cdfoot=&lt;Höhe&gt;</b> .....	<b>29, 23, 30, 31</b>
<b>bleedmargin (tudscrposter)</b> .....	<b>60, 125</b>	<b>cdfoot=nocolor</b> .....	<b>29</b>
		<b>cdfoot=true</b> .....	<b>29</b>
<b>C</b>		<b>cdgeometry</b> .....	<b>23, 24, 33, 34, 55</b>
<b>captions (koma-script)</b>		<b>cdgeometry=adapted</b> .....	<b>22</b>
<b>captions=tableheading</b> .....	<b>107</b>	<b>cdgeometry=calculated</b> .....	<b>23</b>
<b>cd</b> .....	<b>24–27, 32, 41</b>	<b>cdgeometry=custom</b> .....	<b>23, 22</b>
<b>cd=barcolor</b> .....	<b>24</b>	<b>cdgeometry=false</b> .....	<b>22</b>
<b>cd=bicolor</b> .....	<b>24</b>	<b>cdgeometry=restricted</b> .....	<b>22</b>
<b>cd=color</b> .....	<b>24, 34, 40, 71</b>	<b>cdgeometry=symmetric</b> .....	<b>22</b>
<b>cd=false</b> .....	<b>24, 22</b>	<b>cdgeometry=true</b> .....	<b>22</b>
<b>cd=fullcolor</b> .....	<b>24</b>	<b>cdgeometry=twoside</b> .....	<b>22</b>
<b>cd=lightcolor</b> .....	<b>24</b>	<b>cdhead</b> .....	<b>24, 27, 29, 32–34, 60, 73</b>
<b>cd=true</b> .....	<b>24</b>	<b>cdhead=barcolor</b> .....	<b>28</b>
<b>cd (tudscrposter)</b> .....	<b>60</b>	<b>cdhead=bicolor</b> .....	<b>28, 29</b>
<b>cd=barcolor</b> .....	<b>60</b>	<b>cdhead=date</b> .....	<b>28</b>
<b>cd=bicolor</b> .....	<b>60</b>	<b>cdhead=false</b> .....	<b>28, 18</b>
<b>cd=false</b> .....	<b>60</b>	<b>cdhead=heavy</b> .....	<b>28, 18</b>
<b>cd=lightcolor</b> .....	<b>60</b>		

cdhead=lightcolor	28	declaration=false	46
cdhead=nocolor	28	declaration=fill	47, 50
cdhead=nodate	28	declaration=heading	46
cdhead=paperwidth	28	declaration=markboth	47
cdhead=textwidth	28	declaration=multiple	47, 50
cdhead=true	28, 18	declaration=nofill	47, 50
cdmath	19, 73, 105	declaration=nomarkboth	47
cdmath=false	19, 17	declaration=notoc	46
cdmath=true	19, 17	declaration=section	46
cdoldfont	10, 74, 143, 144	declaration=single	47, 50
cdoldfont=din	143	declaration=toc	46
cdoldfont=false	143	declaration=tocleveldown	46
cdoldfont=nodin	143	declaration=tocmultiple	46
cdoldfont=onlydin	143	declaration=true	46
cdoldfont=true	143, 55		
cdpart	25, 60	<b>E</b>	
cdsection	25, 60	extrabottommargin	23, 29–31
cdtitle	25, 60	<b>F</b>	
cdtitle=false	33	fontsize (koma-script)	21, 59, 126, 146
cdtitle=true	33	footlogoheight	23, 30, 31
chapterpage	40, 41	<b>H</b>	
chapterpage=false	40, 26	headings	16, 35
chapterpage=true	40, 26	headings=heavy	25
cleardoublepage (koma-script)	41	headings=light	25, 55
cleardoublespecialpage	33, 40, 41	headings (koma-script)	25
cleardoublespecialpage=color	41	headingsvskip	26
cleardoublespecialpage=current	41	<b>M</b>	
cleardoublespecialpage=false	41	manualmark (sclayer-scrpage)	43, 48
cleardoublespecialpage=nocolor	42	<b>O</b>	
cleardoublespecialpage=<Seitenstil>	41	oldcolors (tudscrcolor)	73
cleardoublespecialpage=true	41	open (koma-script)	
<b>D</b>		open=any	41
ddc	30–34	open=left	22
ddc=black	32	open=right	33, 40, 41
ddc=blue	32	<b>P</b>	
ddc=color	32	pageheadingsvskip	26
ddc=colorblack	32	paper (typearea)	59, 123
ddc=false	31	paper=<Format>	59
ddc=gray	32	paper=<Höhe> :<Breite>	59
ddc=true	31	parskip (koma-script)	15, 44
ddc=white	32	parttitle	
ddcfoot	30–34, 55	parttitle=false	40
ddchead	30–34	parttitle=true	40
declaration	46, 48		
declaration=chapter	46		

<b>R</b>			
relspacing	21, 73	ttfont=false	18
relspacing=false	21, 55	ttfont=lmodern	18, 55
relspacing=true	21	ttfont=true	18
<b>S</b>		tudbookmarks	50, 53, 110
slantedgreek	20, 73	tudbookmarks=false	50
slantedgreek=false	19, 17	tudbookmarks=true	50
slantedgreek=standard	19, 17	tudmathposterfoot (tudscrcomp)	147
slantedgreek=true	19, 17, 55	tudscrver	55
subjectthesis	37, 38, 49, 67	tudscrver=2.03	55
subjectthesis=false	38, 37	tudscrver=2.04	55
subjectthesis=true	38, 37	tudscrver=2.05	55, 10, 74
		tudscrver=2.06	55
		tudscrver=first	55
		tudscrver=last	56, 55
<b>T</b>		twocolumn (typearea)	44, 47, 103
titlepage (koma-script)		twoside (typearea)	
titlepage=false	26, 34, 44	twoside=false	41
titlepage=true	26, 33, 34, 44, 45, 47	twoside=true	22, 29, 33, 40, 41
ttfont	73		

## Befehle und Umgebungen mit zugehörigen Parametern

Dies ist die Auflistung aller im Handbuch erläuterten sowie erwähnten Befehle und Umgebungen. Bei einigen existieren auch Untereinträge für Schlüssel-Wert-Parameter, welche im optionalen Argument der jeweiligen Anweisung verwendet werden können. In der dritten Unterebene sind gegebenenfalls Seitenverweise für explizite Wertzuweisungen an diese Parameter zu finden.

<b>A</b>		<code>\autoref</code> (hyperref)	110, 111, 118
abstract (Umgebung)	43–45, 48, 58, 103	<b>B</b>	
columns	45	<code>\bfseries</code>	17
language	45	<code>\bigskip</code>	21
markboth	44, 45	<code>\blautabelle</code> (tudscrcomp)	148
markboth=false	45	<code>\blocking</code>	46–50, 54
markboth=<Kolumnentitel>	45	company	48, 49, 54
markboth=true	45	<code>\boldmath</code>	19
option	45	<code>\bottomrule</code> (booktabs)	108
pagestyle	45	<b>C</b>	
<code>\addchap</code> (koma-script)	25, 26	<code>\caption</code> (koma-script,caption)	21, 107
<code>\addpart</code> (koma-script)	25, 26	<code>\cdfont</code>	18, 19
<code>\addtokomafont</code> (koma-script)	42, 119	<code>\cefoot</code> (sclayer-scrpage)	27
<code>\advisor</code>	36, 38, 51	<code>\cfoot</code> (sclayer-scrpage)	27
<code>\afterpage</code> (afterpage)	113	<code>\chair</code>	27, 28
<code>\and</code>	25, 35, 36, 38, 49, 51, 54, 61, 67, 87	<code>\chairman</code> (tudscrsupervisor)	64, 68
<code>\author</code>	25, 35, 36, 38, 39, 49, 64, 87	<code>\chapter</code> (koma-script)	25, 26, 41, 43, 46
<code>\author</code> (tudscrposter)	58, 61	<code>\chapterpagestyle</code> (koma-script)	40
<code>\authormore</code>	35	<code>\cmidrule</code> (booktabs)	108
<code>\authormore</code> (tudscrposter)	58, 61		

<code>\cofoot</code> ( <code>sclayer-scrpage</code> )	27	<code>\drittlogofile</code> ( <code>tudscrcomp</code> )	147
<code>\colorbox</code> ( <code>xcolor</code> )	71	<code>\DTMDate</code> ( <code>datetime2</code> )	36, 37, 65
<code>\commaswap</code> ( <code>mathswap</code> )	75	<code>\duedate</code> ( <code>tudscrsupervisor</code> )	64, 65, 68, 103
<code>\company</code>	49, 54	<b>E</b>	
<code>\confirmation</code>	46–50, 54	<code>\emailaddress</code>	35, 61, 67
closing	48, 49	<code>\emailaddress*</code>	35
place	48, 49	<code>\emph</code>	118
supporter	48, 49, 54	<code>\enquote</code> ( <code>csquotes</code> )	103
<code>\confirmation*</code>	49	<code>enumerate</code> (Umgebung) ( <code>koma-script,enumitem</code> )	106
<code>\confirmationclosing</code>	49	<code>evaluation</code> (Umgebung) ( <code>tudscrsupervisor</code> )	91, 36, 65, 66, 68, 69
<code>\contactperson</code> ( <code>tudscrposter</code> )	61	grade	65, 66, 69
<code>\contactperson</code> ( <code>tudscrsupervisor</code> )	66, 67, 69	headline	65, 68
<code>\counterwithin</code>	120	<code>\evaluationform</code> ( <code>tudscrsupervisor</code> )	91, 66, 69
<code>\counterwithout</code>	120	<code>\extraheadline</code>	27, 28
<code>\course</code>	36, 53, 64	<code>\extrarowheight</code> ( <code>array</code> )	108
<code>\course</code> ( <code>tudscrposter</code> )	61	<code>\extratitle</code> ( <code>koma-script</code> )	25, 33, 39, 58
<code>\coverpagebottommargin</code> ( <code>koma-script</code> )	34	<b>F</b>	
<code>\coverpageleftmargin</code> ( <code>koma-script</code> )	34	<code>\faculty</code>	27, 28
<code>\coverpagerightmargin</code> ( <code>koma-script</code> )	34	<code>farbtabellen</code> (Umgebung) ( <code>tudscrcomp</code> )	148
<code>\coverpagetopmargin</code> ( <code>koma-script</code> )	34	<code>figure</code> (Umgebung)	107
<b>D</b>			
<code>\date</code>	25, 28, 36, 37, 49, 52, 66, 103	<code>filecontents</code> (Umgebung) ( <code>filecontents</code> )	112
after	36	<code>\floatsetup</code> ( <code>floatrow</code> )	107
before	36	<code>\footcolumn0</code> ( <code>tudscrcomp</code> )	147
place	37	<code>\footcolumn1</code> ( <code>tudscrcomp</code> )	147
<code>\date*</code>	36	<code>\footcolumn2</code> ( <code>tudscrcomp</code> )	147
<code>\dateofbirth</code>	35, 53, 87, 103	<code>\footcontent</code>	23, 27, 29–31, 61
<code>\declaration</code>	46–48	<code>\footcontent*</code>	31
<code>declarations</code> (Umgebung)	46–50	<code>\footlogo</code>	23, 27, 29, 30, 32–34
columns	48, 49	<code>\footlogosep</code>	30
language	48, 49	<code>\footnote</code> ( <code>koma-script</code> )	39
markboth	47–49	<code>\frontispiece</code> ( <code>koma-script</code> )	25, 33, 39, 58
markboth=false	48	<code>\fusszeile</code> ( <code>tudscrcomp</code> )	147
markboth=<Kolumnentitel>	48	<b>G</b>	
markboth=true	48	<code>\geometry</code> ( <code>geometry</code> )	23, 124
option	48, 49	layout=<Zielformat>	124
pagestyle	48, 49	layoutoffset=<Längenwert>	124
<code>\dedication</code> ( <code>koma-script</code> )	25, 39	paper=<Papierformat>	124
<code>\defensedate</code>	37, 52, 103	showcrop=<Ein-Aus-Wert>	124
<code>\department</code>	27, 28	<code>\getfield</code>	54
<code>description</code> (Umgebung) ( <code>koma-script,enumitem</code> )	106	<code>\grade</code> ( <code>tudscrsupervisor</code> )	69
<code>\discipline</code>	36, 53, 64	<code>\graduation</code>	37, 38, 52
<code>\discipline</code> ( <code>tudscrposter</code> )	61	<code>\grautabelle</code> ( <code>tudscrcomp</code> )	148
<code>\dots</code>	104, 121	<b>H</b>	
<code>\dotswap</code> ( <code>mathswap</code> )	75	<code>\headlogo</code>	27, 30–34

<code>\hypersetup</code> ( <code>hyperref</code> )	35, 62, 125	<b>O</b>	<code>\office</code> ( <code>tudscrposter</code> )	61
<code>\hyphenation</code>	117		<code>\office</code> ( <code>tudscrsupervisor</code> )	67
<b>I</b>			<code>\ofoot</code> ( <code>scrpage</code> )	27
<code>\ifoot</code> ( <code>scrpage</code> )	27		<code>\oldstylenums</code>	17
<code>\ifpdf</code> ( <code>ifpdf</code> )	122	<b>P</b>	<code>\pagestyle</code> ( <code>koma-script</code> )	27, 41
<code>\includegraphics</code> ( <code>graphicx</code> )	30, 67, 100		<code>\paragraph</code> ( <code>koma-script</code> )	16, 26
<code>\institute</code>	27, 28		<code>\part</code> ( <code>koma-script</code> )	25, 26, 40
<code>\institutslogofile</code> ( <code>tudscrcomp</code> )	147		<code>\partpagestyle</code> ( <code>koma-script</code> )	39, 40
<code>\issuedate</code> ( <code>tudscrsupervisor</code> )	64, 65, 68, 103		<code>\PassOptionsToPackage</code>	71, 122
<code>\item</code> ( <code>koma-script,enumitem</code> )	65, 67		<code>\pdfbookmark</code> ( <code>hyperref</code> )	50
<code>itemize</code> ( <code>Umgebung</code> ) ( <code>koma-script,enumitem</code> )	65, 67, 106		<code>\place</code>	37, 49, 66
<b>K</b>			<code>\placeofbirth</code>	36, 53, 87
<code>\KOMAOption</code> ( <code>koma-script</code> )	11, 15		<code>\printdate</code> ( <code>isodate</code> )	36, 37, 65, 103
<code>\KOMAOptions</code> ( <code>koma-script</code> )	11, 15		<code>\professor</code>	38, 51, 64, 65
<b>L</b>			<code>\professor</code> ( <code>tudscrposter</code> )	61
<code>\label</code>	110		<code>\providecaptionname</code> ( <code>koma-script</code> )	51
<code>\leftfoot</code> ( <code>scrpage</code> )	27		<code>pspicture</code> ( <code>Umgebung</code> ) ( <code>pstricks</code> )	110, 122, 123
<code>\rightfoot</code> ( <code>scrpage</code> )	27		<code>\publishers</code> ( <code>koma-script</code> )	25, 34, 39
<code>longtabu</code> ( <code>Umgebung</code> ) ( <code>tabu</code> )	109	<b>Q</b>	<code>quotation</code> ( <code>Umgebung</code> ) ( <code>koma-script</code> )	44, 103
<code>\lowertitleback</code> ( <code>koma-script</code> )	25, 33, 39, 42		<code>quote</code> ( <code>Umgebung</code> ) ( <code>koma-script</code> )	103
<b>M</b>			<code>quoting</code> ( <code>Umgebung</code> ) ( <code>quoting</code> )	45, 103
<code>\makecover</code>	25, 33	<b>R</b>	<code>\raggedtitle</code>	35
<code>\maketitle</code>	25, 32–34, 39, 58		<code>\recalctypearea</code> ( <code>typearea</code> )	119
<code>\maketitleonecolumn</code>	33, 34		<code>\ref</code>	110, 118
<code>\marginnote</code> ( <code>marginnote</code> )	111		<code>\referee</code>	36, 38, 51, 64, 66
<code>\marginpar</code> ( <code>koma-script</code> )	111, 114		<code>\refoot</code> ( <code>scrpage</code> )	27
<code>\matriculationnumber</code>	36, 53, 64, 87		<code>\renewcaptionname</code> ( <code>koma-script</code> )	45, 51
<code>\matriculationyear</code>	36, 53, 64, 87		<code>\rofoot</code> ( <code>scrpage</code> )	27
<code>\mdseries</code>	17	<b>S</b>	<code>\schnittrand</code> ( <code>tudscrcomp</code> )	148
<code>\medskip</code>	21		<code>\section</code> ( <code>koma-script</code> )	26, 43, 46
<code>\midrule</code> ( <code>booktabs</code> )	108		<code>\setchapterpreamble</code> ( <code>koma-script</code> )	40, 41, 45
<code>multicols</code> ( <code>Umgebung</code> ) ( <code>multicol</code> )	59		<code>\setchaptersubtitle</code>	41
<code>\multicolumn</code>	109		<code>\setlist</code> ( <code>enumitem</code> )	106
<code>\multirow</code> ( <code>multirow</code> )	109		<code>\setpartpreamble</code> ( <code>koma-script</code> )	40, 45
<b>N</b>			<code>\setpartsubtitle</code>	40
<code>\newcolumntype</code> ( <code>array</code> )	108		<code>\setstretch</code> ( <code>setspace</code> )	102, 103, 119
<code>\newgeometry</code> ( <code>geometry</code> )	23		<code>\sisetup</code> ( <code>siunitx</code> )	122
<code>\nextabstract</code>	44, 45		<code>\smallskip</code>	21
<code>\nextdeclaration</code>	48		<code>\subfiles</code> ( <code>subfiles</code> )	111
<code>\noindent</code>	120			
<code>notice</code> ( <code>Umgebung</code> ) ( <code>tudscrsupervisor</code> )	93, 36, 66, 69			
headline	66, 69			
<code>\noticeform</code> ( <code>tudscrsupervisor</code> )	93, 67, 68			

<code>\subject</code> ...	25, 34, 37, 38, 49, 58, 67, 68, 87	<code>\textdbn</code> .....	144
<code>\subparagraph</code> (koma-script) .....	16, 26	<code>\textellipsis</code> .....	104, 121
<code>\subsection</code> (koma-script) .....	26	<code>\thanks</code> (koma-script) .....	25, 39
<code>\subsubsection</code> (koma-script) .....	26	<code>\thesis</code> .	34, 35, 37, 38, 49, 64, 65, 67, 68, 87
<code>\subtitle</code> .....	25, 35, 58	<code>\title</code> .....	25, 35, 39, 40, 54, 58, 64
<code>\supervisor</code> .....	36, 38, 49–51, 64	<code>\titledelimiter</code> .....	39, 53
<code>\supporter</code> .....	49, 54	<code>\titlehead</code> (koma-script) .....	25, 39, 58
<b>T</b>		<code>\toprule</code> (booktabs) .....	108
<code>tabbing</code> (Umgebung) .....	108	<code>\tudbookmark</code> .....	50, 110
<code>table</code> (Umgebung) .....	107	<code>\TUOption</code> .....	11, 15, 45, 74, 100
<code>\tableofcontents</code> (koma-script) .....	120	<code>\TUOptions</code> .....	11, 15, 19, 45, 74, 100
<code>tabu</code> (Umgebung) ( <code>tabu</code> ) .....	109	<code>tudpage</code> (Umgebung) .....	24, 32, 64–66, 103
<code>tabular</code> (Umgebung) .....	108, 109	<code>cdhead</code> .....	66
<code>tabular*</code> (Umgebung) .....	108	<code>columns</code> .....	32
<code>tabularx</code> (Umgebung) ( <code>tabularx</code> ) .....	108	<code>language</code> .....	32
<code>tabulary</code> (Umgebung) ( <code>tabulary</code> ) .....	108	<code>pagestyle</code> .....	32
<code>task</code> (Umgebung) ( <code>tudscrsupervisor</code> ) ..	89, 36, 64, 65, 67	<code>\TUDScriptClassName</code> .....	150
<code>headline</code> .....	64, 67	<code>\TUDScriptVersion</code> .....	150
<code>style</code>		<code>\TUDScriptVersionNumber</code> .....	150
<code>style=inline</code> .....	64	<b>U</b>	
<code>style=table</code> .....	64	<code>\uppertitleback</code> (koma-script) ..	25, 33, 39, 42
<code>\taskform</code> ( <code>tudscrsupervisor</code> ) .....	89, 65, 67, 68	<b>W</b>	
<code>\telefax</code> ( <code>tudscrposter</code> ) .....	61	<code>\webpage*</code> ( <code>tudscrposter</code> ) .....	62
<code>\telefax</code> ( <code>tudscrsupervisor</code> ) .....	67	<code>widetable</code> (Umgebung) ( <code>widetable</code> ) .....	108
<code>\telephone</code> ( <code>tudscrposter</code> ) .....	61	<b>X</b>	
<code>\telephone</code> ( <code>tudscrsupervisor</code> ) .....	67	<code>\xspace</code> ( <code>xspace</code> ) .....	104, 121
<code>\textcd</code> .....	18, 19		

## Sprachabhängige Bezeichner

Dies sind alle von TUD-Script definierten, sprachabhängigen Bezeichner. Informationen zur Verwendung sowie den Anpassungsmöglichkeiten sind in Unterkapitel 2.4 zu finden.

<b>A</b>		<b>C</b>	
<code>\abstractname</code> .....	43–45, 53	<code>\chairmanname</code> ( <code>tudscrsupervisor</code> ) .....	64, 68
<code>\advisorname</code> .....	51	<code>\confirmationname</code> .....	46–48, 53
<code>\advisorothername</code> .....	51	<code>\confirmationtext</code> .....	48, 49, 54
<code>\assessmentname</code> ( <code>tudscrsupervisor</code> ) .....	66, 69	<code>\contactname</code> ( <code>tudscrposter</code> ) .....	61, 62
<code>\authorname</code> ( <code>tudscrposter</code> ) .....	61, 62	<code>\contactpersonname</code> ( <code>tudscrposter</code> ) .....	61, 62
<b>B</b>		<code>\contactpersonname</code> ( <code>tudscrsupervisor</code> ) .....	69
<code>\bachelorthesisname</code> .....	38, 52	<code>\contentname</code> ( <code>tudscrsupervisor</code> ) .....	66, 69
<code>\blockingname</code> .....	46–48, 53	<code>\coursename</code> .....	53
<code>\blockingtext</code> .....	49, 54	<code>\coverpagename</code> .....	53



<b>D</b>	
\dateofbirthtext .....	53
\datetext .....	52
\defensedatetext .....	52
\diplomathesisname .....	38, 52
\disciplinename .....	53
\dissertationname .....	38, 52
\duedatetext (tudscrsupervisor) .....	68
<b>E</b>	
\evaluationname (tudscrsupervisor) .....	65, 68
\evaluationtext (tudscrsupervisor) .....	65, 68
<b>F</b>	
\focusname (tudscrsupervisor) .....	64, 67, 68
<b>G</b>	
\gradetext (tudscrsupervisor) .....	66, 69
\graduationtext .....	52
<b>H</b>	
\habilitationname .....	38
<b>I</b>	
\internshipname .....	38, 52
\issuedatetext (tudscrsupervisor) .....	68
<b>L</b>	
\listingname .....	54
\listlistingname .....	54
\logname .....	38, 52
<b>M</b>	
\masterthesisname .....	38, 52
\matriculationnumbername .....	53
\matriculationyearname .....	53
<b>N</b>	
\namesname (tudscrsupervisor) .....	68
\noticename (tudscrsupervisor) .....	66, 69
<b>O</b>	
\objectivesname (tudscrsupervisor) .....	64, 68
<b>P</b>	
\placeofbirthtext .....	53
\professorname .....	51, 64
\professorothername .....	51
\projectpapername .....	38, 52
<b>R</b>	
\refereename .....	51, 64, 66
\refereeothername .....	51, 64, 66
\reportname .....	38, 52
\researchname .....	38, 52
<b>S</b>	
\seminarpapername .....	38, 52
\studentresearchname .....	38, 52
\studentthesisname .....	38
\supervisorname .....	51, 64
\supervisorothername .....	51, 64
<b>T</b>	
\taskname (tudscrsupervisor) .....	64, 66, 67, 69
\tasktext (tudscrsupervisor) .....	64, 67
\termpapername .....	38
\titlename (tudscrsupervisor) .....	68
\titlepagename .....	53

## Seitenstile, Schriftelemente und Farben

<b>A</b>	
author (Schriftelement) (koma-script) .....	35
<b>C</b>	
chaptersubtitle (Schriftelement) .....	41, 42
<b>D</b>	
date (Schriftelement) (koma-script) .....	35
dedication (Schriftelement) (koma-script) .....	35
<b>E</b>	
empty (Seitenstil) .....	27, 60
empty.tudheadings (Seitenstil) .....	27, 60
<b>H</b>	
headings (Seitenstil) .....	27, 45, 48
HKS07 (Farbe) (tudscrcolor) .....	71, 73
HKS33 (Farbe) (tudscrcolor) .....	71, 73
HKS36 (Farbe) (tudscrcolor) .....	71, 73
HKS41 (Farbe) (tudscrcolor) ..	24, 26, 32, 60, 71, 73
HKS44 (Farbe) (tudscrcolor) .....	71, 73
HKS57 (Farbe) (tudscrcolor) .....	71, 73
HKS65 (Farbe) (tudscrcolor) .....	71, 73

HKS92 (Farbe) (tudscrcolor) .....	73	subject (Schriftelement) (koma-script) .....	35
<b>P</b>		subtitle (Schriftelement) (koma-script) .....	35
partsubtitle (Schriftelement) .....	40, 42	<b>T</b>	
plain (Seitenstil) .....	27	thesis (Schriftelement) .....	35, 42
plain.scrheadings (Seitenstil) (sclayer-scrpage)	27	title (Schriftelement) (koma-script) .....	35
plain.tudheadings (Seitenstil) .....	27, 39	titlehead (Schriftelement) (koma-script) .....	35
publishers (Schriftelement) (koma-script) .....	35	titlepage (Schriftelement) .....	35, 42
<b>S</b>		tudheadings (Schriftelement) .....	27, 29, 31, 42
scrheadings (Seitenstil) (sclayer-scrpage) .....	27	tudheadings (Seitenstil) ..	18, 23, 24, 27–32, 45, 76, 100, 125

## Längen und Zähler

<b>A</b>		<b>C</b>	
\abovecaptionskip (Länge) (koma-script) .....	21	\columnsep (Länge) .....	21
<b>B</b>		<b>M</b>	
\belowcaptionskip (Länge) (koma-script) .....	21	\medskipamount (Länge) .....	21
biburllcpentalty (Zähler) (biblatex) .....	119	<b>S</b>	
biburlnumpenalty (Zähler) (biblatex) .....	119	\smallskipamount (Länge) .....	21
biburlucpenalty (Zähler) (biblatex) .....	119	<b>T</b>	
\bigskipamount (Länge) .....	21	\tabcolsep (Länge) .....	109, 119, 120

## Klassen, Pakete und Dateien

<b>A</b>		<b>C</b>	
acro (Paket) .....	107	calc (Paket) .....	112, 146
acronym (Paket) .....	107	caption (Paket) .....	107
afterpage (Paket) .....	113	cfr-lm (Paket) .....	104
amsmath (Paket) .....	146	cleveref (Paket) .....	111
amsmath (Paket) .....	105, 146	cmbright (Paket) .....	136
amssymb (Paket) .....	21, 100	coseoul (Paket) .....	113
array (Paket) .....	108	crop (Paket) .....	112, 101, 124, 125
auto-pst-pdf (Paket) .....	110, 122, 123	csquotes (Paket) .....	103, 106
<b>B</b>		<b>D</b>	
babel (Paket) .....	102, 32, 45, 48, 51, 101, 117	datetime2 (Paket) .....	103, 35–37, 65, 101
balance (Paket) .....	103	dehyph-exptl (Paket) .....	102, 117
beamer (Klasse) .....	10	DIN_Bd_PS.zip (Datei) .....	133, 135
biblatex (Paket) .....	106, 103, 116, 119, 127	DIN_TTF.zip (Datei) .....	133
blindtext (Paket) .....	112, 79	<b>E</b>	
bm (Paket) .....	105	ellipsis (Paket) .....	104, 121
bookmark (Paket) .....	110	enumitem (Paket) .....	106
booktabs (Paket) .....	108	environ (Paket) .....	101, 136

etexcmds (Paket) .....	101	lmodern (Paket) .....	104, 16, 18
etoolbox (Paket) .....	101, 119, 136	longtable (Paket) .....	108
<b>F</b>			
filecontents (Paket) .....	112, 127	ltablex (Paket) .....	108
filemod (Paket) .....	113	ltxtable (Paket) .....	108, 109
fix-cm (Paket) .....	114, 127	lua-check-hyphen (Paket) .....	104
fix-tudscrfonts (Paket) .....	74, 133	<b>M</b>	
flafter (Paket) .....	108, 125	marginnote (Paket) .....	111, 114
float (Paket) .....	113	mathastext (Paket) .....	105, 16, 100
floatrow (Paket) .....	107, 113, 122, 123	mathswap (Paket) .....	75, 98, 105, 150
fnpct (Paket) .....	104	mathswap.pdf (Tutorial) .....	98, 75
fontenc (Paket) .....	101, 102, 113, 117	mathtools (Paket) .....	105
fontinst (Paket) .....	135, 136	mathtype.pdf (Tutorial) .....	98
fontspec (Paket) .....	101, 16, 133, 142	mdsymbol (Paket) .....	16, 21, 100
<b>G</b>			
geometry (Paket) .....	100, 23, 124, 125	microtype (Paket) .....	101, 102
german (Paket) .....	102	miktex-portable.cmd (Datei) .....	136
glossaries (Paket) .....	107, 10, 110, 127, 146	morewrites (Paket) .....	113, 127
graphicx (Paket) .....	100, 30	mparhack (Paket) .....	114, 111
<b>H</b>			
hyperref (Paket) .....	110, 35, 50, 62, 101, 111, 113, 118, 125, 127	mptopdf (Paket) .....	136
hyphsubst (Paket) .....	102, 101, 117	multicol (Paket) .....	103, 21, 32, 45, 48, 59, 101
<b>I</b>			
icomma (Paket) .....	106, 75	multirow (Paket) .....	109
ifpdf (Paket) .....	122	mwe (Paket) .....	112
inkscape.exe (Datei) .....	123	mweights (Paket) .....	100
inputenc (Paket) .....	113	<b>N</b>	
ionumbers (Paket) .....	105, 75	newtx (Paket) .....	105
isodate (Paket) .....	103, 35–37, 65, 101	newtxmath (Paket) .....	105
iwona (Paket) .....	100, 136	ngerman (Paket) .....	102
<b>K</b>			
keyval (Paket) .....	112, 11, 100	noindentafter (Paket) .....	103, 120
koma-script (Paket) .....	100, 136	nomenc (Paket) .....	107, 146
kvoptions (Paket) .....	112	<b>O</b>	
kvsetkeys (Paket) .....	100, 112	opensans (Paket) .....	100, 10, 16, 130
<b>L</b>			
l2kurz.pdf (Datei) .....	3	<b>P</b>	
l2tabu.pdf (Datei) .....	3	pagecolor (Paket) .....	112
letltxmacro (Paket) .....	101, 136	pdfpages (Paket) .....	112
libertine (Paket) .....	105	pdftricks2 (Paket) .....	110
libgreek (Paket) .....	105	pgfkeys (Paket) .....	112
listings (Paket) .....	113, 54, 106	placeins (Paket) .....	107, 125
		polyglossia (Paket) .....	102, 101, 117
		pst-pdf (Paket) .....	110
		pstricks (Paket) .....	109, 16, 110, 111
		<b>Q</b>	
		quoting (Paket) .....	103, 44, 101
		<b>R</b>	
		ragged2e (Paket) .....	103, 101

relsize (Paket) . . . . .	105	tudmathposter (Klasse) . . . . .	144, 10, 74, 146–148
roboto-mono (Paket) . . . . .	18	tudposter (Klasse) . . . . .	74
<b>S</b>			
sansmath (Paket) . . . . .	105	tudscr4lyx.zip (Datei) . . . . .	116
sansmathfonts (Paket) . . . . .	105	tudscr4texstudio.zip (Datei) . . . . .	116
scrartcl (Klasse) . . . . .	12, 42, 100	tudscrartcl (Klasse) 10, 12, 15, 40, 41, 43, 46, 58	
scrbase (Paket) . . . . .	100, 112	tudscrbook (Klasse) . . . . .	10, 12, 15, 22, 33, 40, 41, 43, 46, 53
scrbook (Klasse) . . . . .	12, 42, 43, 100	tudscrcolor (Paket) . . . . .	42, 71, 73, 100, 150
scrextend (Paket) . . . . .	100	tudscrcomp (Paket) . . . . .	133, 144–146, 148, 150
scrhack (Paket) . . . . .	113	tudscrfonts (Paket) . . . . .	15, 73, 74, 150
scrlayer-scrpage (Paket) . . . . .	100, 24, 27, 29, 31	tudscr_fonts_install.bat (Datei) . . . . .	135, 136
scrletter (Paket) . . . . .	10	tudscr_fonts_install.sh (Datei) . . . . .	135
scrfile (Paket) . . . . .	112	tudscr_fonts_install.zip (Datei) . . . . .	135
scrlltr2 (Klasse) . . . . .	10	tudscrposter (Klasse) . . . . .	95, 10, 40, 41, 58, 146, 147
scrreprt (Klasse) . . . . .	12, 42, 100	tudscrreprt (Klasse) . . . . .	10, 12, 15, 22, 41, 43, 46
scrwfile (Paket) . . . . .	113, 127	tudscrsupervisor (Paket) 10, 24, 36, 50, 53, 67,	89, 91, 93, 103, 140, 141, 150
selinput (Paket) . . . . .	113	tudscr_uninstall.bat (Datei) . . . . .	131, 132
selnolig (Paket) . . . . .	104	tudscr_uninstall.sh (Datei) . . . . .	131, 132
setspace (Paket) . . . . .	102, 119	tudscr_v2.06o_install.bat (Datei) . . . . .	132
sfmath (Paket) . . . . .	105	tudscr_v2.06o_install.sh (Datei) . . . . .	132
siunitx (Paket) . . . . .	106, 122	tudscr_v2.06o.zip (Datei) . . . . .	132
spelling (Paket) . . . . .	104	tudthesis (Paket) . . . . .	145
standalone (Paket) . . . . .	111, 110	twocolfix (Paket) . . . . .	75, 150
subcaption (Paket) . . . . .	107	typearea (Paket) . . . . .	100, 22–24, 119
subfig (Paket) . . . . .	107	<b>U</b>	
subfiles (Paket) . . . . .	111, 110	units (Paket) . . . . .	106, 122
svg (Paket) . . . . .	110, 123	Univers_8_TTF.zip (Datei) . . . . .	133
symbols-a4.pdf (Datei) . . . . .	121	Univers_PS.zip (Datei) . . . . .	133, 135
<b>T</b>			
tabu (Paket) . . . . .	109, 108, 119	<b>V</b>	
tabularborder (Paket) . . . . .	109, 119	varioref (Paket) . . . . .	111
tabularx (Paket) . . . . .	108, 109, 119, 146	<b>W</b>	
tabulary (Paket) . . . . .	108, 109, 119	widetable (Paket) . . . . .	108
textcase (Paket) . . . . .	144	<b>X</b>	
textcomp (Paket) . . . . .	105, 146	xcolor (Paket) . . . . .	100, 71, 73, 136
tikz (Paket) . . . . .	109, 73, 110, 111, 123	xkeyval (Paket) . . . . .	112
todonotes (Paket) . . . . .	111, 123, 127	xparse (Paket) . . . . .	112, 122
treatise.pdf (Tutorial) . . . . .	98, 11, 101, 110	xpatch (Paket) . . . . .	101, 136
trimspaces (Paket) . . . . .	101, 136	xpunctuate (Paket) . . . . .	104, 121
tudbeamer (Klasse) . . . . .	10, 74	xspace (Paket) . . . . .	104, 121
tudbook (Klasse) . . . . .	144, 10, 74, 145, 146	<b>Z</b>	
tudfax (Klasse) . . . . .	144, 10, 74, 145	ziffer (Paket) . . . . .	106, 75
tudform (Klasse) . . . . .	144, 10, 74, 145		
tudhaus (Klasse) . . . . .	144, 10, 74, 145		
tudletter (Klasse) . . . . .	144, 10, 74, 145		

# Änderungsliste

# TUD-Script v2.00

## Allgemein

Änderungen gegenüber der vorhergehenden Version .....	139
Schriften – insbesondere für den mathematischen Satz – verbessert .....	15

## Implementierung

abstractdouble=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	139
abstracttoc=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	139
\authortext (Bezeichner): <i>entfällt</i> .....	139
bookmarks=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	140
cd=<Einstellung> (Option): Werte angepasst	
cd=alternative: <i>entfällt</i> .....	139
cdtitle=<Einstellung> (Option): Werte angepasst	
cdtitle=alternative: <i>entfällt</i> .....	139
chapterclear=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	139
color=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	139
\confirmationandrestriction (Befehl): <i>entfällt</i> .....	140
headfoot=<Einstellung> (Option): <i>entfällt</i> .....	139
\location (Befehl): <i>entfällt</i> .....	140
\logfile (Befehl): <i>entfällt</i> .....	139
partclear=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	139
\restrictionandconfirmation (Befehl): <i>entfällt</i> .....	140
\signatureheight (Länge): <i>entfällt</i> .....	140
\titlecoldelim (Bezeichner): <i>entfällt</i> .....	140
\titlecolwidth (Länge): <i>entfällt</i> .....	139
tudfonts=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	139
tudfoot=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	139

## Implementierung tudscrsupervisor (Paket)

\birthday (Befehl): <i>entfällt</i> .....	140
\birthdaytext (Bezeichner): <i>entfällt</i> .....	141
\birthplace (Befehl): <i>entfällt</i> .....	140
\birthplacetext (Bezeichner): <i>entfällt</i> .....	141
cdtask=<Einstellung> (Option): <i>entfällt</i> .....	140
\defensetext (Bezeichner): <i>entfällt</i> .....	141
\duetext (Bezeichner): <i>entfällt</i> .....	141
\enrolmentname (Bezeichner): <i>entfällt</i> .....	141
\enrolmentyear (Befehl): <i>entfällt</i> .....	140
\focustext (Bezeichner): <i>entfällt</i> .....	140
\objectivestext (Bezeichner): <i>entfällt</i> .....	140
\startdate (Befehl): <i>entfällt</i> .....	140
\starttext (Bezeichner): <i>entfällt</i> .....	141
\studentid (Befehl): <i>entfällt</i> .....	140
\studentidname (Bezeichner): <i>entfällt</i> .....	141
\submissiondate (Befehl): <i>entfällt</i> .....	140
\submissiontext (Bezeichner): <i>entfällt</i> .....	141
\supervisorIiname (Bezeichner): <i>entfällt</i> .....	141
\taskcolwidth (Länge): <i>entfällt</i> .....	140

taskcompact=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	140
\tasks (Befehl): <i>entfällt</i> .....	140

## TUD-Script v2.01

### Allgemein

TUD-Script-Bundle auf CTAN veröffentlicht .....	10
Unterschneidung (Kerning) der Ziffern verbessert .....	15

### Implementierung

\maketitle (Befehl): Bugfix für Schriftstärke auf Titelseite .....	33
\subtitle (Befehl): Bugfix für Schriftstärke bei Verwendung des Untertitels .....	35

## TUD-Script v2.02

### Allgemein

Änderungen gegenüber der vorhergehenden Version .....	141
Bugfix für Verwendung der Klasse <b>standalone</b> aus dem Paket <b>standalone</b> .....	111
Installationsroutine der Type1-Schriften angepasst .....	10
OpenType-Schriften mit Paket <b>fontspec</b> verwendbar .....	16
Paket <b>scrlayer-scrpage</b> ist für TUD-Script unabdingbar .....	100
Paketbeschreibung: <b>fontspec</b> .....	101
Paketbeschreibung: <b>listings</b> .....	113
Paketbeschreibung: <b>mwe</b> .....	112
Paketbeschreibung: <b>noindentafter</b> .....	103
Paketbeschreibung: <b>subfiles</b> .....	111
Paketbeschreibung: <b>tabu</b> ist nur bedingt empfehlenswert .....	109
Paketbeschreibung: <b>xkeyval</b> .....	112
Tipps & Tricks: Bezeichnungen der Gliederungsebenen .....	118
Tipps & Tricks: Das Setzen von Auslassungspunkten .....	121
Tipps & Tricks: Fehler beim Schreiben von Hilfsdateien .....	127
Tipps & Tricks: Hinweis auf Rand bei mehrzeiligen Einträgen ergänzt .....	120
Tipps & Tricks: Hinweise zum Paket <b>auto-pst-pdf</b> .....	122
Tipps & Tricks: Literaturverwaltung .....	116
Tipps & Tricks: Lokale Änderungen .....	118
Tipps & Tricks: URL-Umbrüche im Literaturverzeichnis .....	119
Tipps & Tricks: Warnung beim Erzeugen des Inhaltsverzeichnisses .....	120
Tipps & Tricks: Worttrennungen .....	117
<b>tudscrfonts</b> (Paket): <i>neu</i> .....	73
Umbenennung einiger Befehle für Kompatibilität mit anderen Paketen .....	15

### Allgemein **mathswap** (Paket)

Funktionalität im Dokument umschaltbar .....	75
--	----

### Implementierung

abstract=<Einstellung> (Option): Werte angepasst	
abstract=markboth: <i>neu</i> .....	42
abstract=multple: ersetzt abstract=double .....	42
abstract=tocleveldown: <i>neu</i> .....	42

abstract (Umgebung): Parameter angepasst	
markboth=<Einstellung> (Parameter): <i>neu</i> .....	44
pagestyle=<Seitenstil> (Parameter): <i>neu</i> .....	44
abstract (Umgebung): Trennung einzelner Abschnitte mit \nextabstract .....	44
\blocking (Befehl): <i>neu</i> .....	49
\blockingname (Bezeichner): <i>neu</i> .....	53
\blockingtext (Bezeichner): <i>neu</i> .....	54
cdcover=<Einstellung> (Option): <i>neu</i> .....	25
cdfont=<Einstellung> (Option): Werte angepasst	
cdfont=head: <i>neu</i> .....	16
cdfont=heavyhead: <i>neu</i> .....	16
\chapterheadingvskip (Länge): <i>entfällt</i> .....	141
\chapterpagestyle (Befehl): nicht mehr abhängig von \partpagestyle .....	40
\confirmation (Befehl): Parameter angepasst	
columns=<Anzahl> (Parameter): <i>neu</i> .....	48
markboth=<Einstellung> (Parameter): <i>neu</i> .....	48
pagestyle=<Seitenstil> (Parameter): <i>neu</i> .....	48
ddc=<Einstellung> (Option): Logo von DRESDEN-concept automatisch in Kopf/Fuß ....	31
ddc=<Einstellung> (Option): Werte angepasst	
ddc=black: <i>neu</i> .....	31
ddc=blue: <i>neu</i> .....	31
ddc=colorblack: <i>neu</i> .....	31
ddc=gray: <i>neu</i> .....	31
ddc=white: <i>neu</i> .....	31
ddcfoot=<Einstellung> (Option): <i>neu</i> .....	31
ddchead=<Einstellung> (Option): <i>neu</i> .....	31
\declaration (Befehl): Parameter angepasst	
columns=<Anzahl> (Parameter): <i>neu</i> .....	50
markboth=<Einstellung> (Parameter): <i>neu</i> .....	50
pagestyle=<Seitenstil> (Parameter): <i>neu</i> .....	50
declaration=<Einstellung> (Option): Werte angepasst	
declaration=markboth: <i>neu</i> .....	46
declaration=multiple: ersetzt declaration=double .....	46
declaration=tocleveldown: <i>neu</i> .....	46
declarations (Umgebung): <i>neu</i> .....	47
\degree (Befehl): <i>entfällt</i> .....	141
\degreertext (Bezeichner): <i>entfällt</i> .....	141
\discipline (Befehl): <i>neu</i> .....	35
\disciplinename (Bezeichner): <i>neu</i> .....	53
\emailaddress (Befehl): <i>neu</i> .....	35
empty.tudheadings (Seitenstil): <i>neu</i> .....	27
\graduation (Befehl): <i>neu</i> .....	38
\graduationtext (Bezeichner): <i>neu</i> .....	52
\makecover (Befehl): Parameter angepasst	
cdfont=<Einstellung> (Parameter): <i>neu</i> .....	34
pagenumber=<Seitenzahl> (Parameter): <i>neu</i> .....	34
\makecover (Befehl): Umschlagseite für Layout ohne Corporate Design hinzugefügt ....	34



<code>\makecover</code> (Befehl): Unterstützung der Schriftelemente <code>titlehead</code> , <code>subject</code> , <code>title</code> , <code>subtitle</code> , <code>author</code> , <code>publishers</code> , <code>titlepage</code> und <code>thesis</code> .....	34
<code>\maketitle</code> (Befehl): Parameter angepasst	
<code>cdfont=&lt;Einstellung&gt;</code> (Parameter): <i>neu</i> .....	33
<code>pagenumber=&lt;Seitenzahl&gt;</code> (Parameter): <i>neu</i> .....	33
<code>\maketitle</code> (Befehl): Unterstützung der Schriftelemente <code>titlehead</code> , <code>subject</code> , <code>title</code> , <code>subtitle</code> , <code>author</code> , <code>date</code> , <code>publishers</code> , <code>dedication</code> , <code>titlepage</code> und <code>thesis</code> .....	33
<code>\partpagestyle</code> (Befehl): Seitenstil <code>plain.tudheadings</code> wird genutzt .....	39
<code>plain.tudheadings</code> (Seitenstil): <i>neu</i> .....	27
<code>\professorname</code> (Bezeichner): Unterscheidung von einem und mehreren Professoren ...	51
<code>\professorothername</code> (Bezeichner): <i>neu</i> .....	51
<code>\refereename</code> (Bezeichner): Unterscheidung von einem und mehreren Gutachtern .....	51
<code>\restriction</code> (Befehl): <i>entfällt</i> .....	141
<code>\restrictionname</code> (Bezeichner): <i>entfällt</i> .....	141
<code>\restrictiontext</code> (Bezeichner): <i>entfällt</i> .....	141
<code>thesis</code> (Schriftelement): <i>neu</i> .....	35
<code>titlepage</code> (Schriftelement): <i>neu</i> .....	35
<code>tudheadings</code> (Seitenstil): <i>neu</i> .....	27
<code>tudpage</code> (Umgebung): Parameter angepasst	
<code>ddc=&lt;Einstellung&gt;</code> (Parameter): <i>neu</i> .....	32
<code>ddcfoot=&lt;Einstellung&gt;</code> (Parameter): <i>neu</i> .....	32
<code>ddchead=&lt;Einstellung&gt;</code> (Parameter): <i>neu</i> .....	32
<code>foot=&lt;Einstellung&gt;</code> (Parameter): <i>entfällt</i> .....	141
<code>head=&lt;Einstellung&gt;</code> (Parameter): <i>entfällt</i> .....	141
<code>pagestyle=&lt;Einstellung&gt;</code> (Parameter): <i>neu</i> .....	32
Implementierung <b>mathswap</b> (Paket)	
<code>\mathswapoff</code> (Befehl): <i>neu</i> .....	75
<code>\mathswapon</code> (Befehl): <i>neu</i> .....	75
Implementierung <b>tudscrsupervisor</b> (Paket)	
<code>\branch</code> (Befehl): <i>entfällt</i> .....	141
<code>\branchname</code> (Bezeichner): <i>entfällt</i> .....	141
<code>\contact</code> (Befehl): <i>entfällt</i> .....	141
<code>\contactname</code> (Bezeichner): <i>entfällt</i> .....	141
<code>\contactperson</code> (Befehl): <i>neu</i> .....	67
<code>\contactpersonname</code> (Bezeichner): <i>neu</i> .....	69
<code>\email</code> (Befehl): <i>entfällt</i> .....	141
<code>\phone</code> (Befehl): <i>entfällt</i> .....	141
<code>\telephone</code> (Befehl): <i>neu</i> .....	67

## TUD-Script v2.03

### Allgemein

Änderungen gegenüber der vorhergehenden Version .....	142
Anpassungen interner Befehle an KOMA-Script-Version v3.15 .....	15
Bugfix für Umschlagseite und Titel beim Satzspiegel .....	32

Bugfix für Zusammenfassung, Selbstständigkeitserklärung und Sperrvermerk bei der Festlegung von Seitenstil und Kolumnentitel .....	42
Satzspiegel im Corporate Design geändert, Logo von DRESDEN-concept im Fußbereich wird ohne vergrößerten Seitenrand verwendet .....	55
Seitenstile um zweifarbigen Kopf und farbigen Fuß erweitert .....	27
Implementierung	
barfont=<Einstellung> (Option): <i>entfällt</i> .....	142
cd=<Einstellung> (Option): Werte angepasst	
cd=bicolor: Farbeinsatz nur im Kopf mit farbige abgesetztem Querbalken .....	24
cd=fullcolor: voller Farbeinsatz mit farbige abgesetztem Querbalken .....	24
cdchapter=<Einstellung> (Option): Werte angepasst	
cdchapter=bicolor: Farbeinsatz nur im Kopf mit farbige abgesetztem Querbalken ..	26
cdchapter=fullcolor: voller Farbeinsatz mit farbige abgesetztem Querbalken .....	26
cdcover=<Einstellung> (Option): Werte angepasst	
cdcover=bicolor: Farbeinsatz nur im Kopf mit farbige abgesetztem Querbalken ....	25
cdcover=fullcolor: voller Farbeinsatz mit farbige abgesetztem Querbalken .....	25
cdfont=<Einstellung> (Option): Werte angepasst	
cdfont=din: <i>neu</i> .....	16
cdfont=nodin: <i>neu</i> .....	16
cdfonts=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	142
cdfoot=<Einstellung> (Option): Werte angepasst	
cdfoot=color: farbiger Hintergrund der Fußzeile .....	29
cdfoot=<Höhe>: <i>neu</i> .....	29
cdgeometry=<Einstellung> (Option): <i>neu</i> .....	22
cdhead=<Einstellung> (Option): <i>neu</i> .....	27
cdmath=<Ein-Aus-Wert> (Option): <i>neu</i> .....	19
cdpart=<Einstellung> (Option): Werte angepasst	
cdpart=bicolor: Farbeinsatz nur im Kopf mit farbige abgesetztem Querbalken ....	26
cdpart=fullcolor: voller Farbeinsatz mit farbige abgesetztem Querbalken .....	26
cdtitle=<Einstellung> (Option): Werte angepasst	
cdtitle=bicolor: Farbeinsatz nur im Kopf mit farbige abgesetztem Querbalken ....	25
cdtitle=fullcolor: voller Farbeinsatz mit farbige abgesetztem Querbalken .....	25
din=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	142
extrabottommargin=<Höhe> (Option): <i>neu</i> .....	23
\footlogo (Befehl): <i>neu</i> .....	30
\footlogosep (Befehl): <i>neu</i> .....	30
geometry=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	142
\makecover (Befehl): Parameter angepasst	
cdfoot=<Einstellung> (Parameter): <i>neu</i> .....	34
cdhead=<Einstellung> (Parameter): <i>neu</i> .....	34
ddc=<Einstellung> (Parameter): <i>neu</i> .....	34
ddcfoot=<Einstellung> (Parameter): <i>neu</i> .....	34
ddchead=<Einstellung> (Parameter): <i>neu</i> .....	34
footlogo=<Dateinamenliste> (Parameter): <i>neu</i> .....	34
headlogo=<Dateiname> (Parameter): <i>neu</i> .....	34
\maketitle (Befehl): Parameter angepasst	
cdfoot=<Einstellung> (Parameter): <i>neu</i> .....	33

cdhead=<Einstellung> (Parameter): <i>neu</i> .....	33
ddc=<Einstellung> (Parameter): <i>neu</i> .....	33
ddcfoot=<Einstellung> (Parameter): <i>neu</i> .....	33
ddchead=<Einstellung> (Parameter): <i>neu</i> .....	33
footlogo=<Dateinamenliste> (Parameter): <i>neu</i> .....	33
headlogo=<Dateiname> (Parameter): <i>neu</i> .....	33
sansmath=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	142
tudpage (Umgebung): Parameter angepasst	
cdfoot=<Einstellung> (Parameter): <i>neu</i> .....	32
cdhead=<Einstellung> (Parameter): <i>neu</i> .....	32
color=<Farbe> (Parameter): <i>entfällt</i> .....	142
footlogo=<Dateinamenliste> (Parameter): <i>neu</i> .....	32
tudscrver=<Version> (Option): <i>neu</i> .....	55
widehead=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	142

## TUD-Script v2.04

### Allgemein

Änderungen gegenüber der vorhergehenden Version .....	142
Einfachere Verwendung des Paketes <b>fontspec</b> .....	15
Installationsskripte verbessert und robuster gestaltet sowie für die beiden portablen Distributionen <b>T<sub>E</sub>X Live Portable</b> und <b>MiK<sub>T</sub>E<sub>X</sub> Portable</b> erweitert .....	10
Tipps & Tricks: Vermeiden des Skalierens einer PDF Datei beim Druck .....	125
Tipps & Tricks: Warnung bei der Schriftgrößenwahl .....	126
Vertikaler Leerraum abhängig von verwendeter Schriftgröße .....	55

### Allgemein **tudscrfonts** (Paket)

Unterstützung der Klassen <b>tudposter</b> und <b>tudmathposter</b> .....	74
---	----

### Implementierung

abstract=<Einstellung> (Option): Werte angepasst	
abstract=tocmultiple: <i>neu</i> .....	42
cd=<Einstellung> (Option): Werte angepasst	
cd=barcolor: nur farbig abgesetzter Querbalken .....	24
cdchapter=<Einstellung> (Option): Werte angepasst	
cdchapter=barcolor: nur farbig abgesetzter Querbalken .....	26
cdcover=<Einstellung> (Option): Werte angepasst	
cdcover=barcolor: nur farbig abgesetzter Querbalken .....	25
\cdfont (Befehl): <i>neu</i> .....	18
cdhead=<Einstellung> (Option): Werte angepasst	
cdhead=barcolor: nur farbig abgesetzter Querbalken .....	27
cdpart=<Einstellung> (Option): Werte angepasst	
cdpart=barcolor: nur farbig abgesetzter Querbalken .....	26
cdtitle=<Einstellung> (Option): Werte angepasst	
cdtitle=barcolor: nur farbig abgesetzter Querbalken .....	25
declaration=<Einstellung> (Option): Werte angepasst	
declaration=tocmultiple: <i>neu</i> .....	46
declarations (Umgebung): Trennung einzelner Abschnitte mit \nextdeclaration .....	47

fontspec=<Ein-Aus-Wert> (Option): <i>entfällt</i> .....	142
\footcontent (Befehl): <i>neu</i> .....	30
\footcontent* (Befehl): <i>neu</i> .....	30
\textcd (Befehl): <i>neu</i> .....	18
tudheadings (Schriftelement): <i>neu</i> .....	29
\TUDScript (Befehl): <i>neu</i> .....	150
\TUDScriptClassName (Befehl): <i>neu</i> .....	150
\TUDScriptVersion (Befehl): <i>neu</i> .....	150
Implementierung <b>tudscrsupervisor</b> (Paket)	
\namesname (Bezeichner): <i>neu</i> .....	68

## TUD-Script v2.05

### Allgemein

Änderungen gegenüber der vorhergehenden Version .....	142
<b>fix-tudscrfonts</b> (Paket): <i>neu</i> .....	74
Neue Möglichkeiten bei der Satzspiegelberechnung im Corporate Design .....	22
Paketbeschreibung: <b>fnpct</b> .....	103
Paketbeschreibung: <b>scrfile</b> .....	112
Satzspiegeleinstellungen für die jeweilige ISO/DIN Klasse des Papierformates identisch	55
Tipps & Tricks: \includesvg aus Paket <b>svg</b> verwenden .....	123
Tipps & Tricks: Beschnittzugabe und Schnittmarken .....	124
Tipps & Tricks: Fehler beim Laden eines Paketes mit Optionen .....	122
<b>tudscrposter</b> (Klasse): <i>neu</i> .....	58

### Allgemein **tudscrcomp** (Paket)

Unterstützung von <b>tudmathposter</b> .....	146
--	-----

### Allgemein **tudscrfonts** (Paket)

Neues Schriftpaket <b>fix-tudscrfonts</b> für Dokumentklassen im Corporate Design der Technischen Universität Dresden, welche nicht zu TUD-Script gehören .....	74
--	----

### Implementierung

\advisorname (Bezeichner): Unterscheidung von einem und mehreren Fachreferenten ...	51
cdfont=<Einstellung> (Option): Werte angepasst	
cdfont=normalbold: <i>neu</i> .....	16
cdfont=ultrabold: <i>neu</i> .....	16
cdgeometry=<Einstellung> (Option): Werte angepasst	
cdgeometry=adapted: <i>neu</i> .....	22
cdgeometry=calculated: <i>neu</i> .....	22
cdgeometry=custom: <i>neu</i> .....	22
cdgeometry=restricted: <i>neu</i> .....	22
cdhead=<Einstellung> (Option): Werte angepasst	
cdhead=date: Datum zwischen Kopf- und Textbereich .....	27
cdsection=<Einstellung> (Option): <i>neu</i> .....	26
\confirmation* (Befehl): <i>neu</i> .....	48
\course (Befehl): Für Titel verwendbar .....	35
\date (Befehl): Angabe von Parametern für Prä- und Suffix bei Datumsausgabe möglich .	36
\date (Befehl): Parameter angepasst	

after=<Suffix> (Parameter): <i>neu</i> .....	36
before=<Präfix> (Parameter): <i>neu</i> .....	36
place=<Ort> (Parameter): <i>neu</i> .....	36
\date* (Befehl): <i>neu</i> .....	36
\declaration* (Befehl): <i>neu</i> .....	50
\discipline (Befehl): Für Titel verwendbar .....	35
\emailaddress (Befehl): optionales Argument zur Formatierung mit \hypersetup .....	35
\emailaddress* (Befehl): <i>neu</i> .....	35
\footcontent (Befehl): Änderung des Inhaltes nur einer Spalte möglich .....	30
\footlogoheight (Länge): <i>entfällt</i> .....	142
footlogoheight=<Längenwert> (Option): <i>neu</i> .....	30
\headingsvskip (Länge): <i>entfällt</i> .....	142
headingsvskip=<Längenwert> (Option): <i>neu</i> .....	26
\pageheadingsvskip (Länge): <i>entfällt</i> .....	142
pageheadingsvskip=<Längenwert> (Option): <i>neu</i> .....	26
relspacing=<Ein-Aus-Wert> (Option): <i>neu</i> .....	21
\supervisorname (Bezeichner): Unterscheidung von einem und mehreren Betreuern ....	51
\TUDScriptVersionNumber (Befehl): <i>neu</i> .....	150
Implementierung <b>tudscrsupervisor</b> (Paket)	
task (Umgebung): Parameter angepasst	
style=<Stil> (Parameter): <i>neu</i> .....	64
\telefax (Befehl): <i>neu</i> .....	67

## TUD-Script v2.06

### Allgemein

Änderungen gegenüber der vorhergehenden Version .....	143
<b>Open Sans</b> als neue Schrift im Corporate Design der Technischen Universität Dresden	15
Griechische Buchstaben standardmäßig kursiv .....	55
Paket <b>mweights</b> ist für TUD-Script zwingend erforderlich .....	100
Pakete <b>opensans</b> , <b>iwona</b> und <b>mathastext</b> sind für TUD-Script zwingend erforderlich	100
Schriften im Paket <b>opensans</b> , manuelle Installation unnötig .....	10
Verwendung der Roboto Mono als Schreibmaschinenschrift möglich .....	18

### Allgemein **fix-tudscrfonts** (Paket)

<b>Open Sans</b> als neue Hausschrift standardmäßig aktiviert .....	74
---	----

### Allgemein **tudscrfonts** (Paket)

<b>Open Sans</b> als neue Hausschrift standardmäßig aktiviert .....	74
---	----

### Implementierung

cdfont=<Einstellung> (Option): Werte angepasst	
cdfont=din: <i>entfällt</i> .....	144
cdfont=liningfigures: <i>neu</i> .....	16
cdfont=nodin: <i>entfällt</i> .....	144
cdfont=oldstylefigures: <i>neu</i> .....	16
cdfont=standardgreek: <i>neu</i> .....	16
cdoldfont (Option): <i>neu</i> .....	143
chaptersubtitle (Schriftelement): <i>neu</i> .....	41

<code>clearcolor=&lt;Ein-Aus-Wert&gt;</code> (Option): <i>entfällt</i> .....	144
<code>cleardoublespecialpage=&lt;Einstellung&gt;</code> (Option): Einstellung für farbige Rückseiten ..	41
<code>\dinbn</code> (Befehl): <i>entfällt</i> .....	143
<code>\emailaddress</code> (Befehl): wird für alle Klassen bereitgestellt, Verwendung im Titel möglich	35
<code>\emailaddress*</code> (Befehl): wird für alle Klassen bereitgestellt, Verwendung im Titel möglich	35
<code>\frontispiece</code> (Befehl): Schmutztitelrückseite aus <b>koma-script</b> wird unterstützt .....	39
<code>\getfield</code> (Befehl): <i>neu</i> .....	42
<code>headings=&lt;Einstellung&gt;</code> (Option): <i>neu</i> .....	25
<code>\ifdin</code> (Befehl): <i>entfällt</i> .....	144
<code>\maketitle</code> (Befehl): Parameter angepasst	
<code>cdgeometry=&lt;Einstellung&gt;</code> (Parameter): <i>neu</i> .....	33
<code>partsubtitle</code> (Schriftelement): <i>neu</i> .....	40
<code>\raggedtitle</code> (Befehl): <i>neu</i> .....	35
<code>\setchaptersubtitle</code> (Befehl): <i>neu</i> .....	41
<code>\setpartsubtitle</code> (Befehl): <i>neu</i> .....	40
<code>slantedgreek=&lt;Ein-Aus-Wert&gt;</code> (Option): Werte angepasst	
<code>slantedgreek=standard</code> : <i>neu</i> .....	19
<code>\textdbn</code> (Befehl): <i>entfällt</i> .....	143
<code>\textubn</code> (Befehl): <i>entfällt</i> .....	143
<code>\textubs</code> (Befehl): <i>entfällt</i> .....	143
<code>\textuln</code> (Befehl): <i>entfällt</i> .....	143
<code>\textuls</code> (Befehl): <i>entfällt</i> .....	143
<code>\texturn</code> (Befehl): <i>entfällt</i> .....	143
<code>\texturs</code> (Befehl): <i>entfällt</i> .....	143
<code>\textuxn</code> (Befehl): <i>entfällt</i> .....	143
<code>\textuxs</code> (Befehl): <i>entfällt</i> .....	143
<code>titlesignature=&lt;Ein-Aus-Wert&gt;</code> (Option): <i>neu</i> .....	38
<code>ttfont=&lt;Einstellung&gt;</code> (Option): <i>neu</i> .....	18
<code>\univbn</code> (Befehl): <i>entfällt</i> .....	143
<code>\univbs</code> (Befehl): <i>entfällt</i> .....	143
<code>\univln</code> (Befehl): <i>entfällt</i> .....	143
<code>\univls</code> (Befehl): <i>entfällt</i> .....	143
<code>\univrn</code> (Befehl): <i>entfällt</i> .....	143
<code>\univrs</code> (Befehl): <i>entfällt</i> .....	143
<code>\univxn</code> (Befehl): <i>entfällt</i> .....	143
<code>\univxs</code> (Befehl): <i>entfällt</i> .....	143