Die Document-Style-Familie „Script“

Version 2.0

1. Dezember 1993

Haftungsausschluß:
Es wird keinerlei Haftung übernommen für irgendwelche Schäden, die aus der Benutzung der Programme des hier beschriebenen Paketes folgen.

Inhaltsverzeichnis

1. Allgemeines

1.1 Einleitung

Ursprünglich versuchte ich im Rahmen eines größeren Projektes[1] lediglich den „boldface extended"-Font (`cmbx`) in Überschriften durch einen schöneren zu ersetzen und forschte deshalb in den Style-Dateien nach Änderungsmöglichkeiten. Ungefähr gleichzeitig bekam ich ein sehr gutes Buch über Typographie[2] in die Hand, und es wurde mir sehr bald klar: es müssen komplett neue Styles her, die auch die europäischen typographischen Regeln und Papierformate (insbesondere DIN A4) berücksichtigen.

Nach einigen Wochen Arbeit habe ich am 6. Juni 1992 die Version 1.0 der Document-Style-Familie „Script" per FTP veröffentlicht und war sehr überrascht über die große Resonanz. Neben vielen kleineren Problemen (die allerdings meistens auf LaTeX und nicht auf `script` zurückzuführen waren) wurde auch eine echte Unzulänglichkeit zur Verbesserung an mich herangetragen: der `script_l`-Style lieferte in Verbindung mit dem neuen Fontauswahlschema von Mittelbach und Schöpf einige Fehlermeldungen. Dieses Schema – im folgenden kurz NFSS genannt – war damals auf meinem Rechner noch nicht installiert, so daß mir dieses Manko nicht aufgefallen war. Die jetzt vorliegende „Script"-Version 2.0 sollte nun in allen Teilen sowohl mit dem alten als auch dem neuen Fontauswahlschema fehlerfrei laufen. Übrigens kann ich das NFSS besten Gewissens jedermann empfehlen; es ist sehr angenehm damit zu arbeiten und wird ab der nächsten LaTeX-Version sowieso Standard.

Der Erstbenutzer von „Script" kann einen ersten Eindruck von den durchgeführten Änderungen gegenüber den Standardstyles durch diese Dokumentation selbst bekommen; sie wurde mit `\documentstyle[twoside,german]{script}`

---

[1] es handelte sich um ein Vorlesungs-Skript – daher auch der Name für die hier beschriebene Style-Familie

[2] Jan Tschichold: Ausgewählte Aufsätze über die Gestalt des Buches und der Typographie, Zweite Auflage, Birkhäuser Verlag Basel 1987

und nachträglicher Einstellung von `\headseplineon` (siehe S. 15) gesetzt. Doch auch „alten Hasen" sei die Lektüre ans Herz gelegt, da sich so manche Kleinigkeit gegenüber Version 1.0 geändert hat. Diese Modifikationen sind jedoch aufwärtskompatibel und im Text gesondert erwähnt.

1.2 Autor, Copyright und Danksagungen

Die Document-Style-Familie `script` und die dazugehörige Dokumentation (die Sie gerade lesen) wurden von Frank Neukam geschrieben. „Script" ist, wie bei TEX üblich, frei kopierbar (public domain). Nachträgliche Änderungen Ihrerseits sind natürlich wie bei den Originalen legitim, nur bitte ich darum, bei Minimalmodifikationen der Styles die Autorenangabe nicht allzu dreist zu verändern (oder gar meinen werten Namen frech durch den eigenen zu ersetzen!), sondern Kommentarzeilen wie „Modified by . . . " einzufügen. Außerdem sollte dann in der typeout message und der ersten Kommentarzeile das Datum aktualisiert und jede Änderung kommentiert werden.

Sollten Sie Ihre modifizierte Version veröffentlichen wollen, so bitte ich, diese nicht (wie schon geschehen) als „offizielle" neue Ausgabe mit neuer Versionsnummer zu verbreiten und womöglich sogar das alte Original auf einer Mailbox oder einem FTP-Server zu überschreiben. Bitte kennzeichnen Sie in diesem Fall Ihre Version eindeutig als „Patch", belassen Sie die Versionsnummer (sonst gibt es irgendwann zig verschiedene Modifikationen mit identischen Nummern) und speichern Sie sie in einem anderen Verzeichnis oder Archiv als das Original. Falls Sie Ihre Änderungen gerne in einer neuen „offiziellen" Version von `script` sehen würden, so setzen Sie sich bitte mit mir in Verbindung (Adresse siehe unten).

Mein besonderer Dank gilt an dieser Stelle dem Rechenzentrum der Universität Karlsruhe für viele Probeausdrucke, Thorsten Hoeppner, Klaus-Jürgen Wald und Christian Wurll für Ideen und Kritik, Uta Janssen für typographische Beratung, Dr. Karlheinz Fingerle für die Bereitstellung des NFSS, Oliver Schurr für seinen Artikel über „Script" im DANTE-Magazin[3], Markus Kohm für das Patch zur Papiergrößeneinstellung, Sven Gohlke für Programmierarbeit und Tips bezüglich der Sprachumschaltung beim `script_l.sty` sowie allen eifrigen Briefeschreibern, deren Lob, Kritik und Verbesserungsvorschlägen Sie diese neue Version zu verdanken haben.

[3] Die TEXnische Komödie, Heft 3/92

1.3 Was ist was

Das komplette Paket der Version 2.0 besteht aus folgenden Files:

`checkprt.tex`	Testseite zur genauen Einstellung von Druckertreibern
`dir.tex`	Programm zum Drucken eines Adreßbuches aus einem `.adr`-File (siehe S. 29)
`phone.tex`	Programm zum Drucken eines Telefonnummern-Verzeichnisses aus einem `.adr`-File; erst seit Version 2.0 Bestandteil des Paketes
`script.dvi`	Dokumentation zur „Script“-Stylefamilie
`script.sty`	Style-Hauptdatei; entspricht einem Mittelding aus `report.sty` und `book.sty`
`script.tex`	Quelltext zu `script.dvi`
`script10.sty`	Hilfsdatei für alle drei Hauptstyles; wird von diesen bei Einstellung der 10pt-Schrift nachgeladen
`script11.sty`	Dito, für 11 pt
`script12.sty`	Dito, für 12 pt
`script_l.sty`	Style-Hauptdatei „script_letter“; entspricht dem `letter.sty`
`script_s.sty`	Style-Hauptdatei „script_small“; entspricht dem `article.sty`
`sctitle.sty`	Pendant zu `titlepage.sty`; erzeugt eine eigene Titelseite
`typearea.sty`	Style-Option zur einfacheren Satzspiegeleinstellung

1.4 Kompatibilität zu den Originalstyles

Alle Befehle, die in den zugrundeliegenden Original-Styles `article`, `report`, `book` und `letter` vorhanden sind, sind auch weiterhin verfügbar. Es wurden lediglich Befehle hinzugefügt bzw. schon vorhandene Einstellungen und Befehlsdefinitionen modifiziert. Dadurch muß man bei Verwendung der „Script“-Styles eigentlich nichts besonders beachten, nur das Ergebnis wird etwas anders aussehen als gewohnt. So ist es auch kein Problem, „alte“ Texte dem neuen Layout anzupassen, indem man in der Definition des Document-Styles z.B. `article` einfach durch `script_s` ersetzt. Um die neuen Möglichkeiten jedoch voll ausschöpfen zu können, ist die Lektüre dieser Anleitung unumgänglich.

1.5 Fehler

Als Entschädigung dafür, daß sich mit Sicherheit irgendwo neue Fehler in die Styles eingeschlichen haben, wurden ein paar bei den Originalen vorhandene entfernt. Falls Sie einen Fehler finden, der eindeutig auf eine Fehlprogrammierung der verwendeten Style-Dateien zurückzuführen ist, bitte ich Sie, mir das auch im Interesse der anderen Benutzer zu schreiben, damit ich ihn beim nächsten Update entfernen kann. Aber auch für sonstige Kritik und Verbesserungsvorschläge bin ich immer dankbar; bitte bestehen Sie jedoch nicht auf einer Rückantwort und bestellen Sie keine Styles bei mir (irgendwoher müssen Sie diese Anleitung ja haben, da sind auch die Styles selbst bestimmt nicht weit). Hier meine Adresse:

Frank Neukam
Franz-Stadelmayer-Str. 30
97074 Würzburg

2. Das neue Layout

2.1 Die neuen Fonts

Überall, wo früher Extended Boldface verwendet wurde (also in allen Überschriften, im Inhaltsverzeichnis und im `description` Environment) wird jetzt standardmäßig ein neuer Font benutzt, nämlich Sansserif Bold Extended (`cmssbx`). Es handelt sich dabei um eine serifenlose Schrift in halbfetter Ausführung, die durch den Befehl `\sfb`[1] selektiert wird. Zu beachten ist lediglich, daß als kleinste Größe ein 10pt-Font zur Verfügung steht; wird z.B. mittels `\tiny\sfb` eine kleinere Schrift gewünscht, so wird aus Lesbarkeitsgründen die normale Extended Boldface eingestellt, da diese auch noch in kleineren Entwurfsgrößen vorhanden ist.

2.2 Der Satzspiegel

2.2.1 Grundlegendes

Der Satzspiegel definiert die Größe des bedruckten Bereichs einer Seite[2] und dessen Positionierung auf der Seite; seine Breite ist abhängig von der gewählten Schriftart und -größe (optimale Lesbarkeit ist bei ca. 60-70 Zeichen pro Zeile gegeben). Bei seiner Konstruktion ist besonders auf die Seitenproportionen (Verhältnis von Breite zu Höhe) sowie auf seinen Stand auf der Seite zu achten.

Jan Tschichold hat in dem in der Einleitung angegebenen Buch eine Konstruktion vorgeschlagen, die einfach durchführbar ist und bei jedem Seitenverhältnis des gewählten Papierformats (also auch bei dem „unklassischen" DIN A4-Format

[1] Bei Verwendung des NFSS ist die Befehlsfolge `\sf\bf` dem `\sfb`-Kommando vorzuziehen, da `\sfb` aus Kompatibilitätsgründen wie ein „alter" Fontselektionsbefehl arbeitet, also *alle* Fontparameter einstellt.

[2] Im folgenden auch „Textbereich" genannt; Kopfzeile, Fußzeile und Randnoten zählen jedoch nicht dazu.

mit dem Seitenverhältnis 1:$\sqrt{2}$, von dem in allen `script`-Hauptstyles normalerweise ausgegangen wird) sehr ansprechende Ergebnisse liefert. Dies liegt wohl in der Hauptsache daran, daß hier der Satzspiegel die gleichen Seitenverhältnisse wie die zu bedruckende Seite selbst aufweist; auch die Ränder stehen in einer sehr einfachen Relation zueinander (innerer zu äußerer bzw. oberer zu unterer Rand verhalten sich wie 1:2; bei einseitigem Druck, also ohne die `twoside`-Option, sind innerer und äußerer Rand gleich groß).

2.2.2 Konstruktion

Die Konstruktion basiert auf einer Einteilung der Breite und Höhe der Seite in n gleiche Teile. Das Ergebnis ist je eine Längenangabe für die Horizontale – im folgenden mit HLE (horizontale Längeneinheit) abgekürzt – und die Vertikale (VLE).

Beispiel: Für eine DIN A4-Seite (210 mm $\times$ 297 mm) ergibt sich bei Zehnteilung der Seite ($n = 10$) für die horizontale Längeneinheit HLE = 210 mm $\div$ 10 = 21 mm und für die vertikale Längeneinheit VLE = 297 mm $\div$ 10 = 29.7 mm.

Nun wird bei doppelseitigem Druck einfach der innere Rand[3] auf eine Breite von 1 HLE und der äußere Rand auf 2 HLE gesetzt. Soll einseitig gedruckt werden, so wird sowohl der innere als auch der äußere Rand auf 1.5 HLE eingestellt. Analog dazu wird ein oberer Rand von 1 VLE und ein unterer Rand von 2 VLE freigelassen. Für den Textbereich bleibt so eine Fläche von $(n-3)$ HLE Breite mal $(n-3)$ VLE Höhe. Randbemerkungen („marginal notes“) stehen im äußeren Rand und sind auf eine Breite von 1.5 HLE (bei Verwendung der `twoside`-Option) bzw. 1 HLE (sonst) begrenzt.

Beispiel: Bei Sechsteilung der Seite ($n = 6$) sind der linke und rechte Rand zusammen 3 HLE und der Textbereich ebenfalls $6 - 3 = 3$ HLE breit. Das gleiche gilt für die vertikalen Größen. Damit ist bei einer Sechsteilung der Seite der Textbereich genau halb so breit und hoch wie die gesamte Seite.

Die TeX-Forderung, daß die Textbereichsbreite (`\textwidth`) ein Vielfaches der Zeichenbreite der `Typewriter`-Schrift sein soll, wird durch die beschriebene exakte Konstruktion nicht erfüllt. Sie wird allerdings selbst von den Original-Styles nicht eingehalten und ist nur bei komplett in `Typewriter`-Schrift geschriebenen

[3] Es ist hier der effektiv sichtbare Rand gemeint, nicht die odd- bzw. evensidemargin-Einstellung; diese Größen werden intern so errechnet, daß eben genau der gewünschte Rand sichtbar ist.

Zeilen von Belang. In Abschnitt 2.2.4 wird eine Möglichkeit aufgezeigt, wie z.B. für Programmlistings der Satzspiegel doch noch der `\tt`-Schrift angepaßt werden kann.

Die zweite Regel der Satzspiegelkonstruktion, daß `\textheight` (die Höhe des Textbereiches) gleich `\topskip` vermehrt um ein ganzzahliges Vielfaches des Grundlinienabstandes (`\baselineskip`) sein soll, wurde jedoch befolgt. Damit wird erreicht, daß bei komplett mit Fließtext gefüllten Seiten (also Seiten ohne Überschriften, Tabellen, *displayed material* o.ä.) die Absätze nicht auseinandergezogen werden müssen, um zu gewährleisten, daß die unterste Zeile mit dem unteren Rand des Textbereichs zusammenfällt. Der Satzspiegel kann dadurch maximal eine knappe Zeile höher werden als dies bei einer exakten Konstruktion der Fall gewesen wäre.

2.2.3 Auswahl des gewünschten Satzspiegels

Man erkennt leicht, daß für kleine n auch der Textbereich im Verhältnis zur Seitengröße klein ist, und für sehr große n der Textbereich die Seite nahezu ausfüllt (Randbemerkungen sind bei n größer als 12 mangels Platz kaum noch möglich bzw. sinnvoll umbrechbar). So läßt sich mit lediglich einer Größe der Satzspiegel in weiten Grenzen verändern, und man hat außerdem noch die Garantie, daß er immer optimal konstruiert ist.

Optional läßt sich noch ein Bindungs-Korrektur-Maß *BCOR* (für binding-correction) angeben, so daß bei der Satzspiegelkonstruktion berücksichtigt wird, daß ein Teil der Seite durch Heftung oder Bindung nicht sichtbar ist oder abgefräst wird. Die Seite wird dazu um *BCOR* schmäler betrachtet, als sie wirklich ist, der Satzspiegel auf dieser Basis berechnet, und anschließend der innere Rand um dieses Stück vergrößert.

Beispiel: Ist durch eine Klebebindung, bei der sich die Seiten nicht mehr bis ganz innen aufschlagen lassen, der sichtbare Bereich der Seite nicht mehr 210 mm breit (DIN A4), sondern 8 mm weniger, so gibt man für *BCOR* 8 mm an (wie und wo steht weiter unten) und schon steht der Satzspiegel z.B. bei einseitigem Druck wieder exakt mittig auf dem sichtbaren Teil der Seite und hat auch die gleichen Proportionen wie dieser (202:297 statt 210:297).

Der Satzspiegel wird durch Angabe des Befehls

`\typearea[`*BCOR*`]{`*DIV*`}`

neu konstruiert. Weil interne Größen wie `\oddsidemargin` u.ä. verändert werden, die nur in der Präambel (dem Teil der Textquelldatei vor `\begin{document}`)

Textfontgröße	*DIV*
10pt	8
11pt	10
12pt	12

Tabelle 2.1: *DIV*-Voreinstellungen

	Textbereich		Ränder	
DIV	Breite [mm]	Höhe [mm]	oben [mm]	innen [mm]
6	105.00	148.50	49.50	35.00
7	120.00	169.71	42.43	30.00
8	131.25	185.63	37.13	26.25
9	140.00	198.00	33.00	23.33
10	147.00	207.90	29.70	21.00
11	152.73	216.00	27.00	19.09
12	157.50	222.75	24.75	17.50
13	161.54	228.46	22.85	16.15
14	165.00	233.36	21.21	15.00
15	168.00	237.60	19.80	14.00

Tabelle 2.2: Satzspiegelmaße in Abhängigkeit von *DIV*

modifiziert werden dürfen, ist auch der `\typearea`-Befehl nur in der Präambel erlaubt. Die Optionalangabe *BCOR* stellt die gleichnamige Größe zur Bindungs-Korrektur ein, während *DIV* die Rolle des oben genannten Divisors n übernimmt. *BCOR* bedarf einer Maßeinheit, wobei alle bei TEX erlaubten Einheiten möglich sind (pt, mm, cm, in, ...), *DIV* hingegen muß eine ganze Zahl größer als 3 sein.

Bei Verwendung des `script` oder `script_s`-Styles wird *DIV* nach Tabelle 2.1 gewählt (wobei keine Bindungs-Korrektur vorgenommen wird) und der `\typearea`-Befehl durch den `\documentstyle`-Befehl automatisch ausgeführt.

Als Anhaltspunkt für die manuelle *DIV*-Einstellung kann Tabelle 2.2 dienen. Dabei wurde $BCOR = 0$ und das Papierformat DIN A4 sowie zweiseitiger Druck zugrundegelegt; die Abstimmung der Textbereichshöhe auf den (von der gewählten Schriftgröße abhängigen) Grundlinienabstand, die die Textbereichshöhe nochmals um maximal 14.5 pt $\simeq$ 5.1 mm vergrößern kann, wurde nicht berücksichtigt.

2.2.4 Ungewöhnliche Satzspiegel

Wird ein Satzspiegel gewünscht, der ungewöhnliche Proportionen hat oder zwar wie oben angegeben konstruiert werden soll, jedoch *exakt* die in der vorstehenden Tabelle aufgeführten Maße (ohne die Korrektur auf Vielfache des Grundlinienabstandes) haben soll, so gibt es noch alternativ zu `\typearea` den Befehl

`\areaset[`*BCOR*`]{`*WIDTH*`}{`*HEIGHT*`}`

der die Textbereichsbreite auf *WIDTH* und die Textbereichshöhe auf *HEIGHT* setzt (jeweils ohne jede nachträgliche Korrektur) und die Rand- und Randnoten-Einstellungen so vornimmt, daß der Textbereich sinnvoll positioniert wird (ähnlich wie beim `\typearea`-Befehl). Auch die Wirkung der optionalen *BCOR*-Angabe ist wie schon oben beschrieben.

Damit ist es nun sehr einfach möglich, z.B. genau quadratische oder sehr schmale hohe Satzspiegel zu verwirklichen. Wenn die automatische Positionierung nicht gefällt, kann diese durch die entsprechenden Befehle natürlich auch noch nachträglich verändert werden. Weiterhin ist es hiermit möglich, die Satzspiegelbreite auf die Breite eines vorgegebenen Textes abzustimmen. Als Beispiel soll, wie weiter oben angekündigt, der Textbereich der Breite von 30 Zeichen in `Typewriter`-Schrift angepaßt werden. Eine Befehlsfolge, die dieses durchführt, würde etwa lauten:

```
\newlength{\thirtytt}
\settowidth{\thirtytt}{\tt 123456789012345678901234567890}
\areaset{\thirtytt}{20cm} % Textbereichsh"ohe: 20cm
```

Auch dieser Befehl darf selbstverständlich nur in der Präambel auftreten.

2.2.5 Berücksichtigung von Kopf- und/oder Fußzeile

Manchmal ist es sinnvoll, daß bei der Konstruktion des Satzspiegels auch die Kopf- und/oder Fußzeile als zum Textbereich dazugehörig betrachtet wird. Dies ist z.B. der Fall, wenn der Kolumnentitel wie in dieser Dokumentation durch eine horizontale Linie vom eigentlichen Text getrennt ist und dadurch optisch näher an diesen heranrückt.

Bei Angabe des `\headincludeon` Befehls gilt bei einer nachfolgenden Satzspiegelberechnung mit Hilfe des `\typearea`- oder `\areaset`-Befehls alles, was oben über den Textbereich gesagt wurde, für Textbereich plus Kopfzeile. Dieses Verhalten läßt sich natürlich mit der komplementären Anweisung `\headincludeoff` wieder abschalten. Für Fußzeilen gibt es ab der Version 2.0 von `script` die entsprechenden Befehle `\footincludeon` bzw. `\footincludeoff`.

2.2.6 Andere Papierformate

Sollen sich die Satzspiegelberechnungen nicht auf DIN A4 (wie es hier voreingestellt ist), sondern auf ein anderes Papierformat beziehen, so muß nun nicht mehr – wie dies noch in der Version 1.0 der „Script"-Styles der Fall war – die Datei `typearea.sty` modifiziert werden. Ab Version 2.0 lassen sich andere Blattmaße als DIN A4 einfach durch die Änderung der Variablen `\paperwidth` bzw. `\paperheight` (vor dem Aufruf des `\typearea`- bzw. `\areaset`-Befehls) einstellen. Einem Vorschlag von Markus Kohm folgend, werden diese beiden Variablen automatisch durch die Angabe einer der Optionen `DINA3`, `DINA4`[4], `DINA5`, `DINA6`, `DINA3l`, `DINA4l`, `DINA5l` oder `DINA6l` im `\documentstyle` Befehl gesetzt und der Satzspiegel entsprechend berechnet. Der Buchstabe `l` am Ende dieser Optionen kennzeichnet dabei die Verwendung des Querformats (`l` steht für „Landscape", dem englischen Fachbegriff für Druck im Querformat).

Beispiel 1: Ein Gedichtbändchen soll im DIN A5-Format gedruckt werden. Dazu genügt nun die Angabe von `DINA5` als Styleoption, z.B.:

```
\documentstyle[german,DINA5]{script}
```

Beispiel 2: Der Textbereich eines Zeitplanes im DIN A4 *Quer*format soll exakt 270 mm mal 190 mm groß sein. Dies wird erreicht durch die Angabe von

```
\documentstyle[german,DINA4l]{script_s}
\areaset{270mm}{190mm}
```

und einer (systemabhängigen) Anweisung an den Druckertreiber, im Querformat zu drucken.

Beispiel 3: Beim Gedichtbändchen aus Beispiel 1 soll zusätzlich eine Abfräsung von 1,5 mm berücksichtigt werden. Dazu muß der Satzspiegel nochmal neu konstruiert werden, z.B. mit

```
\documentstyle[german,DINA5]{script}
\typearea[1.5mm]{12}
```

Beispiel 4: Es soll auf Endlospapier mit den Maßen $8\frac{1}{4}$ inch mal 12 inch gedruckt werden, und auch dort soll der Satzspiegel ordentlich aussehen. Mit den Befehlen

[4]Diese Option wurde nur der Vollständigkeit halber mit aufgenommen, da auch ohne deren Angabe das Papierformat DIN A4 zugrunde gelegt wird.

```
\paperwidth 8.25in
\paperheight 12in
\typearea{10}
```

wird dies erreicht. Druckt man öfters auf Endlospapier, so empfiehlt es sich, diese Anweisungen in eine eigene Optionsdatei (z.B. `endlos.sty`) zu schreiben, die man dann einfach z.B. mit

```
\documentstyle[endlos,german]{script}
```

verwenden kann. Soll zusätzlich das Argument des `\typearea`-Befehls abhängig von der gewählten Schriftgröße sein, so ersetzt man in dieser Optionsdatei `\typearea{10}` durch

```
\ifcase \@ptsize\relax
 \typearea{8}      %10pt
 \or \typearea{10} %11pt
 \or \typearea{12} %12pt
\fi
```

2.2.7 Einbindung der Befehle zur Satzspiegeleinstellung

In die beiden Hauptstyles `script` und `script_s` sind die Befehle `\typearea` und `\areaset` bereits eingebunden. Da der brieferzeugende `script_l`-Style einen besonderen und vor allem festen Satzspiegel besitzt, wurde bei ihm auf die Einbindung verzichtet.

Um diese Befehle auch in anderen Styles benutzen zu können, wurden ihre Definitionen zusätzlich seperat in die Optionsdatei `typearea.sty` geschrieben. Diese nimmt auch die von der gewählten Schriftgröße abhängigen Voreinstellungen des Satzspiegels vor. Bei Verwendung des originalen `article`-Styles wird also durch Angabe von

```
\documentstyle[11pt,typearea]{article}
```

die 11pt-Schrift ausgewählt und anschließend der neu eingebundene `\typearea{10}`-Befehl automatisch ausgeführt.

Selbstverständlich kann `typearea.sty` auch in Verbindung mit jedem anderen Style benutzt werden, jedoch ist dieses nur sinnvoll, wenn dieser nicht von einem festen Layout ausgeht (wie dies z.B. beim `letter`-Style und seinen Varianten der Fall ist).

2.3 Sonstige Änderungen

2.3.1 Default-Fontgröße

Wird in der Definition des Document-Styles nichts anderes angegeben, so wird standardmäßig als Textfont der 11pt-Font verwendet (wie z.B. in dieser Dokumentation). Eine Ausnahme hiervon bildet wieder der `script_l`-Style, bei dem ohne Optionsangabe die 12pt-Schrift benutzt wird. Im Gegensatz zu den Originalstyles wird jedoch keine Fehlermeldung erzeugt, wenn der Default-Font trotzdem als Option angegeben wird (probieren Sie doch einmal `\documentstyle[10pt]{article}`...).

2.3.2 Kapitelüberschriften und Nummerierung

Kapitelüberschriften im `script`-Style werden ohne die bei `report` und `book` üblichen eigenen Zeilen für das Wort „Kapitel" und die Kapitelnummer gesetzt, sondern wie bei Überschriften niedrigerer Priorität mit der Kapitelnummer direkt vor der eigentlichen Überschrift. Da hierdurch ein Anhang leicht für ein normales Kapitel gehalten werden kann (er unterscheidet sich von diesen ja nur durch die Art der Nummerierung), ist es empfehlenswert, ihn mit `\part*{Anhang}` deutlich vom Rest des Dokumentes abzusetzen.

Nach einfachen Abschnittsnummern steht einheitlich ein Punkt (1. statt 1 , aber nicht 1.1. sondern 1.1). Die einzige Ausnahme bildet die Nummerierung der Parts, da bei römischer Zählung ein nachfolgender Punkt unüblich ist. Dies gilt selbstverständlich auch für Einträge in Kolumnentitel und in das Inhaltsverzeichnis.

2.3.3 Kolumnentitel und Pagina

Der `script`-Style benutzt als Default den Pagestyle `headings`, der in den Kopfteil der Seiten das aktuelle Kapitel (linke Seiten) bzw. den aktuellen Abschnitt (rechte Seiten) schreibt (sog. lebender Kolumnentitel bzw. running head). Weiterhin wird eine Seitennummer in den Fußbereich der Seite gesetzt (sog. Pagina). `script_s` stellt standardmäßig `plain` als Pagestyle ein, jedoch sind natürlich auch hier `headings`, `myheadings` und `empty` verfügbar. Beim Brief stehen teilweise andere Seitenstile zur Auswahl, die in Abschnitt 4.2.7 behandelt werden.

Normalerweise werden Kolumnentitel und Pagina zentriert; wurde jedoch die `twoside` Option angegeben, so wird beides an den äußeren Rand der Seite gesetzt.

Die Definitionen der Pagestyles `headings`, `myheadings` und `plain`, sowie die Befehle zum automatischen Setzen der Kolumnentitel durch Befehle wie `\section` oder `\tableofcontents` wurden dahingehend verändert, daß die Kolumnentitel mit normaler Groß-Klein-Schreibung (vorher alles groß), standardmäßig in *Italics* (vorher *slanted*) und bei Kapiteln ohne das Wort CHAPTER bzw. KAPITEL gesetzt werden. Die Seitennummer wurde generell aus der Kopfzeile in die sonst leere Fußzeile verbannt. Diese Maßnahmen haben neben der besseren Lesbarkeit („Inhaltsverzeichnis" liest sich viel angenehmer als „INHALTSVERZEICHNIS") vor allem den Vorteil, daß nun auch wesentlich längere Abschnitts-Überschriften in die Kopfzeile passen ohne abgeschnitten zu werden.

Wird unter den Kolumnentiteln eine horizontale Linie gewünscht, so ist auch dies jetzt ohne Probleme möglich: sie läßt sich einfach mit

```
\headseplineon
```

an- bzw. durch

```
\headseplineoff
```

wieder ausschalten. Das funktioniert natürlich nur bei den Pagestyles `headings` und `myheadings`, da ja alle anderen keinen Kolumnentitel generieren. Durch diese Befehle wird auch `\headinclude` automatisch sinnvoll miteingestellt, d.h. `\headincludeon` bei `\headseplineon` und `\headincludeoff` bei `\headseplineoff`. Damit dies zum Tragen kommt, muß allerdings nach einer Veränderung dieses Schalters der Satzspiegel neu konstruiert werden. Default ist in den beiden Hauptstyles `script` und `script_s` (bei `script_l` ist wieder alles anders; siehe Abschnitt 4.2.7) `\headseplineoff`.

Beispiel: In der vorliegenden Dokumentation wurde mit `\headseplineon` der horizontale Strich unter den Kolumnentiteln eingeschaltet (dadurch wurde auch gleichzeitig `\headincludeon` ausgeführt). Damit der Satzspiegel dieser Änderung angepaßt wird, mußte dieser mit `\typearea{10}` neu berechnet werden.

Ab der Version 2.0 von `script` und `script_s` gilt das oben für eine horizontale Linie unter den Kolumnentiteln gesagte auch für die Fußzeile. Mit

```
\footseplineon
```

kann man die Erzeugung einer solche Linie oberhalb der Seitennummer in den Pagestyles `plain`, `headings` und `myheadings` an-, und mit

```
\footseplineoff
```

wieder abschalten. Analog zu den `\headinclude`-Befehlen existieren auch die Varianten für die Fußzeile `\footincludeon` und `\footincludeoff`.

2.3.4 Kleinere kosmetische Veränderungen

In Fußnoten ist ein kleiner horizontaler Bereich (1 em) nur für die Fußnotennummer reserviert, der eigentliche Text beginnt also immer erst diesen Abstand vom linken Rand entfernt. Bei mehrzeiligen Fußnoten werden die Folgezeilen nochmals 0.5 em links eingezogen. Dadurch setzen sich diese besser voneinander ab.

Absätze werden ohne vertikalen Zwischenraum und mit einem Einzug von 1 em (entspricht 1 `\quad`) gesetzt.

Zusätzlich wurde der im europäischen Raum übliche „französische" Punktbehandlungsmodus (`\frenchspacing`) selektiert, was den Vorteil hat, daß man nicht mehr hinter Abkürzungen, die mit einem Kleinbuchstaben enden, ein `\␣` schreiben muß, um einen zu großen Zwischenraum zu vermeiden.

Bei der `enumerate`-Umgebung wurde in der zweiten Ebene eine Klammer entfernt, so daß a) statt (a) geschrieben wird. Weiterhin wurde in der dritten Ebene des `itemize`-Environments der „$*$" durch ein „$\triangleright$" ersetzt (das sieht zumindest für meinen Geschmack etwas gefälliger aus).

Die Autoren- und Datumsangaben in Titeln erscheinen nicht in `\large`, sondern in `\Large`.

In mehrzeiligen Captions (Unterschriften von `table` bzw. `figure`) werden die zweiten und folgenden Zeilen so weit links eingerückt, daß diese nicht mehr unter „Abbildung . . . :", sondern unter dem eigentlichen Text der ersten Zeile beginnen.

Es wurden noch etliche weitere Kleinigkeiten im Layout verändert, die allerdings dem ungeübten Auge kaum auffallen werden. So ist etwa der Spaltenabstand im zweispaltigen Satz von 10 pt auf die übliche 1 cc vergößert worden (1 Cicero $\simeq$ 12 pt). Will man alles ganz genau wissen, so kann man ja immer noch die jeweilige Style-Datei studieren (dazu sollte man allerdings auch die niedrigeren TEX-Befehle kennen bzw. zuerst die `.doc`-Versionen der Originalstyles durcharbeiten).

3. Zusätzliche Befehle

3.1 Die Erweiterungen des Abstracts und der Titelseite

Bei Abstracts besteht nun die Möglichkeit, die automatisch erzeugte Überschrift (z.B. „Zusammenfassung“) mit `\abstractoff` aus- bzw. mit `\abstracton` wieder anzuschalten (voreingestellt ist `\abstracton`).

Der `\maketitle`-Befehl des `script`-Hauptstyles wurde durch die folgenden optionalen Angaben wesentlich erweitert:

- `\extratitle{`*Schmutztitel*`}` erzeugt eine eigene Seite, auf der der sog. „Schmutztitel“[1] steht. Dieser wird wie normaler Text, also ohne Zentrierung und in normaler Schriftart und -größe gesetzt. Dadurch ist dessen Gestaltung völlig dem Anwender überlassen. Die Rückseite ist auch im zweiseitigen Druck leer.

- In den Kopfteil der Haupttitelseite kann nun mittels `\titlehead` zusätzlicher Text wie z.B. eine Institutsadresse geschrieben werden.

- Direkt oberhalb des eigentlichen Titels wird durch `\subject` eine Zusatzangabe (z.B. „Diplomarbeit“) in etwas größerer Schrift eingefügt.

- `\publishers{`*Verlag*`}` plaziert den Verlagsnamen in den unteren Teil der Haupttitelseite.

- Im zweiseitigen Druck läßt sich die Rückseite des Haupttitels für weitere Informationen (im Beispiel dieser Dokumentation ist dies der Haf-

[1] Früher war der Buchblock oftmals nicht durch einen Buchdeckel vor Verschmutzung geschützt; diese Aufgabe übernahm dann die erste Seite des Buches, die meist einen Kurztitel, eben jenen „Schmutztitel“, trug. Auch heute noch wird gerne eine Extraseite vor dem eigentlichen Haupttitel für Verlagsname, Buchreihennummer o.ä. benutzt.

tungsausschluß) nutzen. Dazu stehen die Befehle `\uppertitleback` und `\lowertitleback` zur Verfügung.

- `\dedication{`*Widmung*`}` erzeugt eine seperate Widmungsseite, wobei *Widmung* zentriert und in etwas größerer Schrift gesetzt wird. Wie bei `\extratitle` ist die Rückseite grundsätzlich leer.

Beispiel:

```
\titlehead{Institut f"ur Nachrichtentechnik \hfill SS1993
         \\Hochschulstr.12
         \\3456 Unistadt}
\subject{Diplomarbeit}
\title{Digitale Raumsimulation mit dem DSP56004}
\author{Ulrich Ungenannt}
\lowertitleback{Diese Diplomarbeit wurde mit dem Textsatzsystem
        \TeX\ erstellt.}
\maketitle
```

Der Titel (Kurz- oder Haupttitel) beginnt normalerweise auf Seite 1; im Original war hierfür Seite 0 vorgesehen, was zu der unangenehmen Tatsache führte, daß die Titelseite im zweiseitigen Druck links zu liegen kam. Durch Angabe einer Zusatzoption (z.B. `\maketitle[-1]`) läßt sich die Startseite jetzt beliebig festlegen.

Im `script_s`-Hauptstyle ist von den genannten Features lediglich `\titlehead` und `\subject` vorhanden. Um jedoch auch dort oder in anderen Styles wie z.B. `article` eine eigene Titelseite mit allen zusätzlichen Möglichkeiten zur Verfügung zu stellen, wurden alle dazu notwendigen Definitionen in die Optionsdatei `sctitle.sty` geschrieben. Ab Version 2.0 dieses Files ist jedoch zu beachten, daß der Font, in dem der Haupttitel und der Titel des Abstracts erscheint, vom Hauptstyle abhängig ist: Bei Verwendung von `script_s`, Version 2.0, werden die Titel mit dem Zeichensatz gesetzt, der im `\sectfont`-Befehl definiert ist (siehe Abschnitt 3.2). In Verbindung mit allen Hauptstyles, die diesen Befehl nicht kennen (das dürften fast alle anderen sein, z.B. `article`, `report`, `book`, `script_s` Version 1.0 usw.), wird für die Titel `\bf` selektiert.

3.2 Die „xxxxfont“-Variablen

Seit der Version 2.0 des „Script“-Paketes sind mehrere Variablen zur globalen Schriftänderung in Kolumnentiteln, Paginierung, Captions, Überschriften und dem `description`-Environment bereitgestellt. Dadurch lassen sich mit einem einzigen Befehl z.B. sämtliche Überschriften statt in `\sfb` in `\rm` ausgegeben. Dazu muß nur (am besten in der Präambel) die Anweisung

```
\def\sectfont{\rm}
```

oder, was gleichbedeutend ist,

```
\renewcommand{\sectfont}{\rm}
```

stehen. Auf die gleiche Art und Weise können auch die anderen Fonteinstellungen geändert werden:

Variable	Wirkung auf:	Default
`\captfont`	Caption-Texte	`\rm`
`\descfont`	Item-Texte des `description`-Environments	`\sfb`
`\headfont`	Kolumnentitel	`\it`
`\pnumfont`	Paginierung	`\rm`
`\sectfont`	Sämtliche Überschriften und Titel	`\sfb`

Selbstverständlich sind nicht nur Fonteinstellungs-, sondern auch Größenänderungsbefehle möglich. Im `script_l`-Style ist keiner dieser Variablen definiert, da es dort weder Captions noch Überschriften gibt und Kopf- sowie Fußzeilen häufig grundlegend modifiziert werden.

3.3 Die „marginline“

Analog zum `\marginpar` wurde die `\marginline` eingeführt, die für einzeilige Randnotizen gedacht ist. Bei diesen tritt nämlich im `twoside`-Modus auf linken Seiten das Problem auf, daß sie mit `\marginpar` nicht rechtsbündig gesetzt werden (was sinnvoll wäre), sondern durch den Blocksatz einer nicht vollen Zeile linksbündig erscheinen. Bei Verwendung von `\marginline{`*Randnotiz*`}` werden einzeilige Texte sinnvoll in den äußeren Rand gesetzt, was zum Beispiel zum Hervorheben von Querverweisen nützlich ist.

3.4 Der „addchap“- bzw. „addsec“-Befehl

Für unnummerierte Abschnitte wie z.B. ein Vorwort oder Danksagungen stand bisher nur der `\chapter*`- bzw. `\section*`-Befehl zur Verfügung, die allerdings den

Nachteil haben, daß sie zwar das bis auf die Nummerierung komplette Überschrifts-Layout erzeugen, jedoch weder einen Eintrag ins Inhaltsverzeichnis noch einen Kolumnentitel generieren. Dies mußte, falls dies erwünscht war, nachträglich von Hand durchgeführt werden, was wohl nicht ganz der LaTeX-Philosophie entspricht.

Gerade für die oben genannten Anwendungen wurden daher die neuen Befehle `\addchap` („additional chapter", bei `script.sty`) und `\addsec` („additional section", bei `script_s.sty`) implementiert, die bis auf die fehlende Nummerierung genauso wie der `\chapter`- bzw. `\section`-Befehl (ohne Stern!) arbeiten, also alle Einträge an den entsprechenden Stellen erzeugen. Natürlich existieren auch `\addchap*` bzw. `\addsec*`, die den Stern-Versionen der üblichen Befehle sehr ähnlich sind, nur daß zusätzlich die Kolumnentitel gelöscht werden (geht z.B. der zu einem `\chapter*` gehörende Text über mehrere Seiten, so tauchen die – veralteten! – Kolumnentitel plötzlich wieder auf). Zusätzlich ist als Optionalargument wie bei den Originalen ein anderer Eintrag ins Inhaltsverzeichnis als die Abschnittsüberschrift angebbar.

3.5 Der „minisec"-Befehl

Manchmal ist eine Art Überschrift wünschenswert, die zwar etwas hervortritt (etwa durch die Wahl einer anderen Schrift), ansonsten aber eng mit dem nachfolgenden Text zusammenhängt, also nicht wie bei den üblichen Sektionierungsbefehlen relativ viel vertikalen Freiraum vor und nach der Überschrift läßt.

Wirkung des neuen „minisec"-Befehls:
Der Befehl `\minisec` („minisec" steht für „mini-section") bewirkt nun genau das oben genannte, indem er das Argument in `\sfb` mit einem vertikalen Freiraum von 1.5 ex vorher und keinem danach setzt. Die obige Zwischenüberschrift wurde als Beispiel mit `\minisec{Wirkung des ...}` gesetzt. Mini-Sections werden nicht mit ins Inhaltsverzeichnis aufgenommen. Da von vorneherein keine Nummerierung vorgesehen ist, existiert auch keine Version mit Stern von diesem Befehl. Genauso wie `\marginline`, `\addchap`, `\addsec` und die Titelseitenerweiterungen existiert `\minisec` nicht im `script_l`-Style.

3.6 Das „labeling"-Environment

Wirkung: Wie in dieser tabellarischen Aufstellung, die auch gleichzeitig wieder ein Beispiel für die Wirkung des Environments ist, setzt „labeling" Text wie in einem normalen Absatz, nur daß zusätzlich ein Titel vorangestellt werden kann (in diesem Fall „Wirkung").

Pro Titel sind, wie hier vorgeführt, mehrere Absätze möglich, um auch umfangreichere Texte übersichtlich gliedern zu können.

Syntax: „labeling" ist ein gewöhnliches Environment mit einer Zusatzangabe plus einem Optionalargument:

```
\begin{labeling}[Trennzeichen]{längster Titel}
\item[Titel] ...
...
\end{labeling}
```

Dabei ist *längster Titel* das längste `\item`-Argument inklusive allen eventuellen Schriftänderungsbefehlen. Wird kein Trennzeichen benötigt, so kann dieses zusammen mit dem eckigen Klammerpaar weggelassen werden.

Beispiel: Die Standardform eines mathematischen Beweises wird folgendermaßen erzeugt:

```
\begin{labeling}[:]{\sfb Auss.}
\item[\sfb Vor.] $x$ sei ...
\item[\sfb Auss.] Es gilt ...
\item[\sfb Bew.] Nach Satz ...
\end{labeling}
```

Dies bewirkt die Ausgabe:

Vor. : x sei ...

Auss.: Es gilt ...

Bew. : Nach Satz ...

4. Der script_l-Style

4.1 Das Original

Nur wenige LaTeX-Anwender haben bisher auch Briefe mit TeX geschrieben, was wohl vor allem daran liegt, daß der originale `letter`-Style sehr auf das amerikanische Briefformat zugeschnitten und außerdem schlecht dokumentiert ist. Daher soll hier kurz der Aufbau eines Quelltextes für Briefe erläutert werden. Ein typisches Eingabefile hat folgende Form:

```
\documentstyle[german]{letter}
\name{Name des Absenders}
\address{Adresse des Absenders}
\signature{Unterschrift}
\begin{document}

\begin{letter}{Name und Adresse des Empf"angers des ersten Briefes}
\opening{Anrede (z.B. Sehr geehrte Damen und Herren,)}
Brieftext...Brieftext...Brieftext
\closing{Gru"sformel (z.B. Mit freundlichen Gr"u"sen)}
\ps{Postskriptum}
\cc{Verteiler (z.B. Direktion\\Personalb"uro)}
\encl{Anlagen (z.B. Lebenslauf\\Lichtbild)}
\end{letter}

\begin{letter}{Name und Adresse des Empf"angers des zweiten Briefes}
   ...
\end{letter}

   ...
\end{document}
```

Die Befehle `\signature`, `\closing`, `\ps`, `\cc` und `\encl` sind optional. Bei fehlender `\signature`-Angabe wird das `\name`-Argument als Unterschrift gesetzt. Ist eine gedruckte Unterschrift völlig unerwünscht, so kann diese durch `\signature{\ }` unterdrückt werden.

Zu beachten ist weiterhin, daß Adreßfeld, Datum usw. erst durch den `\opening`-Befehl erzeugt werden (wichtig z.B. bei Rechnungen ohne Anrede; dort muß dann trotzdem ein leerer `\opening{}`-Befehl stehen). Dies hat weiterhin zur Folge, daß z.B. bei einer zu langen `backaddress` (s.u.) eine „Overfull hbox" in der Zeile gemeldet wird, in der der `\opening`-Befehl steht, obwohl dieser gar nicht die eigentliche Ursache ist. Bei einer solchen Warnung sind also auch alle vorherigen Befehle zu prüfen!

Um sich bei jedem Brief die wiederholte Eingabe der eigenen Adresse und sonstiger Voreinstellungen zu ersparen, ist es empfehlenswert, diese in eine eigene Datei (z.B. `privat.sty`) zu schreiben, die dann einfach als Option beim `\documentstyle` angegeben wird (ohne `.sty`). Ein Beispiel für eine solche Datei findet sich auf Seite 29.

4.2 Zusätzliche Eigenschaften des script_l-Styles

Der neue `script_l`-Hauptstyle ist gegenüber dem Original wesentlich erweitert und flexibler gestaltet worden. Ihn zeichnet vor allem die Möglichkeit der Benutzung eigener Adreßdateien und nahezu uneingeschränkte Serienbrieffähigkeit aus. Im Gegensatz zum `letter`-Style, der die fontgrößenabhängigen Einstellungen selbst vornimmt, verwendet `script_l` die Hilfsdateien `script10.sty`, `script11.sty` und `script12.sty`.

4.2.1 Mehrsprachigkeit

Der `script_l`-Style wurde von vorneherein mehrsprachig ausgelegt (englisch, amerikanisches englisch, deutsch, österreichisch und französisch; voreingestellt ist amerikanisches englisch). Seit der Version 2.0 von `script_l` wird die Mehrsprachigkeit von TEX3 voll unterstützt; dies hat allerdings zur Folge, daß die Benutzung eines „antiken" TEX (älter als Version 3) in Verbindung mit dem neuen Briefstyle eine Fehlermeldung mit Abbruchmöglichkeit produziert.

Zur Einstellung der Sprache der automatisch gesetzten („Caption"-) Texte wie „Betreff", „Seite", „Anlagen" usw. stehen die folgenden Befehle zur Verfügung:

```
\captionsenglish
\captionsUSenglish
```

```
\captionsgerman
\captionsaustrian
\captionsfrench
```

Die Schreibweise des Datums wird wird durch die nachstehenden Befehle festgelegt:

`\dateenglish` : 1/12/1993

`\dateUSenglish` : 12/1/1993

`\dategerman` : 1. 12. 1993

`\dateaustrian` : 1. 12. 1993

`\datefrench` : 1. 12. 1993

Es ist generell empfehlenswert (aber nicht zwingend notwendig), grundsätzlich `german` oder `german3` als Option im `\documentstyle`-Befehl anzugeben, da diese Styles den sehr komfortablen `\selectlanguage`-Befehl zur Verfügung stellen, der gleichzeitig Captiontexte, die Schreibweise des Datums, einige sprachspezifische Extras und Trennregeln der verwendeten Sprache anpaßt. Im Zusammenhang mit `script_l` ist jedoch zu beachten, daß `\selectlanguage` erst *nach* `\begin{document}` verwendet werden sollte[1].

Beispiel 1: Für eine Datei mit rein englischsprachigen Briefen (amerikanisches Englisch) genügt:

```
\documentstyle{script_l}
\begin{document}
\begin{letter}{...}
(...)
```

Beispiel 2: Für eine Datei mit rein deutschsprachigen Briefen genügt:

```
\documentstyle[german]{script_l}
\begin{document}
\begin{letter}{...}
(...)
```

[1]Die briefspezifischen Caption-Texte und Datumsformate werden nach einem Vorschlag von Sven Gohlke erst durch den `\begin{document}`-Befehl definiert. Daher werden diese erst nach diesem Befehl aktiv.

Beispiel 3: Bei gemischtsprachigen Briefen empfiehlt sich:

```
\documentstyle[german]{script_l}
\begin{document}

\selectlanguage{USenglish}
\begin{letter}{...}
(...)
\end{letter}

\selectlanguage{german}
\begin{letter}{...}
(...)
\end{letter}

(...)
```

4.2.2 Zweiseitiger Druck

Zweiseitig gedruckte Briefe werden durch Angabe der `twoside`-Option unterstützt; jedoch ändert sich hier im Gegensatz zum `script`- oder `script_s`-Style der Satzspiegel nicht, sondern es wird lediglich sichergestellt, daß ein neuer Brief immer auf einer ungeraden *Druck*seite beginnt. Die *interne* TEX-Seitenzählung startet bei jedem `\begin{letter}`-Befehl wieder bei 1.

4.2.3 Layout

Der `script_l`-Style verwendet einen festen Satzspiegel, an dem nichts mehr verändert werden sollte. Er ist recht anspruchsvoll, was Maßhaltigkeit und bedruckbaren Bereich des verwendeten Druckers angeht, da Falzmarken so weit links wie möglich und die Adresse wirklich im Adreßfenster eines Fensterbriefumschlages stehen sollen. Aus diesem Grund wurde dem Paket eine Datei `checkprt.tex` mitgegeben, die eine Testseite zur Drucker(treiber)einstellung erzeugt. Mit ihrer Hilfe kann sehr leicht in Millimetern und Inch gemessen werden, welcher Bereich der Seite bedruckbar ist und ob die Offsets des `dvi`-Treibers stimmen.

Die Höhe des Kopfbereichs der Seite ist in der `script_l`-Version 2.0 von 20 mm auf 22 mm vergrößert worden[2]; damit gibt es hoffentlich keine Probleme mehr bei Verwendung des voreingestellten Briefkopf-Formates[3]. Der Fußbereich der Seite ist 20 mm hoch, linker und rechter Rand sind beide 25.4 mm breit.

Absätze werden ohne Einzug und mit einem vertikalen Abstand von einem halben `\baselineskip` gesetzt. Ein unterer Randausgleich wird auch im zweiseitigen Druck nicht durchgeführt (`\raggedbottom`).

Gegenüber der Version 1.0 wurde die Brieftext-Zentrierung leicht verbessert, was die Probleme bei umfangreicheren Briefen mit sehr langen Absätzen hoffentlich beseitigt hat.

4.2.4 Adreßfeld und Zusatz-„location"-Feld

Zur Beschriftung des Briefumschlages gab es im Original den `\makelabels`-Befehl, der dazu gedacht war, Adreßetiketten zu bedrucken. Da mir keine Konfiguration bekannt ist, in der dieser Befehl mit all seinen Anforderungen bezüglich Papierwechsel etc. wirklich Sinn macht, habe ich ihn erst gar nicht in den neuen Briefstyle miteinbezogen (mal ehrlich: haben Sie `\makelabels` benutzt?) Hier sehe ich auch die – hoffentlich – einzige Möglichkeit, wo eine Inkompatibilität zu den Originalstyles auftreten kann.

Als Entschädigung ermöglicht `script_l` die Verwendung von Fensterbriefumschlägen, die Adreßetiketten überflüssig machen. In einem Feld der Höhe `\addrheight` und Breite `\addrwidth`, das `\addrvskip` von der Textbereichsoberkante und (`\addrindent` plus 25.4 mm) vom linken Rand des Papiers entfernt ist, wird dazu die bei `\begin{letter}` angegebene Adresse vertikal zentriert. Optional kann noch mit

`\backaddress{`*Rücksende-Adresse*`}`

eine kleingeschriebene Rücksende-Adresse und mit

`\specialmail{`*Versandart*`}`

eine spezielle Versandart (z.B. Einschreiben) in dieses Feld eingeblendet werden.

[2] Dadurch ist der Abstand des oberen Papierrandes vom Briefkopf von 10 mm auf 9 mm geschrumpft; die meisten mir bekannten Laser- und Tintenstrahldrucker kommen jedoch auch damit noch zurecht. Sollten durch diese Änderung Probleme entstehen, so können diese durch geeignete Einstellung des Druckertreibers in der Regel umgangen werden.

[3] Diese Probleme traten auf bei Verwendung der 12pt-Schrift und wenn beide Zeilen der Adresse Unterlängen besaßen. Dreizeilige Adressangaben sind jedoch nach wie vor mangels Platz nicht möglich. Falls mehr Information im Briefkopf untergebracht werden soll, so muß ein eigener entwickelt werden.

Im freien Raum rechts neben dem Adreßfeld läßt sich noch zusätzlicher Text (z.B. Zweigstellen einer Firma) unterbringen. Dieser wird durch

`\location{`*Zusatz-Information*`}`

in ein `\locwidth` breites Feld geschrieben, dessen rechter Rand mit der rechten Begrenzung des Textbereiches übereinstimmt.

Beide Felder zusammen sind durch `\addrfieldoff` ab- bzw. mit `\addrfieldon` wieder anschaltbar (Default: `\addrfieldon`). Dies kann z.B. für persönliche Schreiben ohne Geschäftsbriefcharakter von Nutzen sein (wenngleich ich in diesem Falle eher zum Füller greifen würde).

Noch eine kleine Anmerkung zur Schreibweise des Wortes „Adresse“: deutsch schreibt es sich mit *einem* „d“, während es im Englischen „a*dd*ress“ heißt. Daher ist bei allen Befehlen, die „address“ enthalten, auf das doppelte „d“ zu achten. Anweisungen, die etwas mit dem unten beschriebenen `.adr`-File zu tun haben, werden allerdings mit *einem* „d“ geschrieben (`\adrentry` und `\adrchar`).

4.2.5 Referenzfeld

Unterhalb des Adreßfeldes steht normalerweise das Datum bzw. bei Geschäftsbriefen ein Referenzfeld mit vorgedruckten Unterfeldern für Kundennummer, Durchwahl, Sachbearbeiter und ähnliches. Dies ist jetzt auch möglich: ohne weitere Angabe erscheint rechtsbündig Ort (definiert durch `\place{`*Ort*`}`) und Datum, während bei Benutzung mindestens eines der folgenden Befehle das Datum in Vordruckform und die gewünschten Unterfelder horizontal gegeneinander zentriert gesetzt werden:

`\yourref{`*Ihr Zeichen*`}`
`\yourmail{`*Ihr Schreiben vom*`}`
`\myref{`*Unser Zeichen*`}`
`\customer{`*Kundennummer*`}`
`\invoice{`*Rechnungsnummer*`}`
`\refitemi{`*Eigenes Feld 1*`}`
`\refitemii{`*Eigenes Feld 2*`}`
`\refitemiii{`*Eigenes Feld 3*`}`

Die „Vordrucke“ der selbstdefinierten Unterfelder lassen sich mit `\refitemnamei`, `\refitemnameii` und `\refitemnameiii` bestimmen.

4.2.6 Titel und Betreff

Keine eigene Titelseite, sondern eine fettgedruckte Überschrift (z.B. „Rechnung“, „1. Mahnung“) wird durch

`\title{`*Überschrift*`}`

unterhalb des Referenzfeldes erzeugt.
Ein „Betreff“ läßt sich mit

`\subject{`*Kurzinhalt des Briefes*`}`

angeben, wobei mit `\subjecton` bzw. `\subjectoff` das Wort „Betr.“ selbst an- und ausgeschaltet werden kann (voreingestellt ist `\subjectoff`).

4.2.7 Seitenstile

Im `script_l`-Style stehen nach wie vor die Seitenstile `empty`, `plain` und `headings` zur Verfügung (Default ist `plain`). Zusätzlich gibt es noch speziell für die erste Seite eines Briefes den Pagestyle `firstpage`. Der eigentliche Clou an der Sache ist nun, daß die Einträge in den Kopf- und Fußbereich der Seiten bei `firstpage` und `headings` frei gestaltbar sind. Dies geschieht durch die Befehle `\firsthead`, `\firstfoot`, `\nexthead` und `\nextfoot`, die bei einer Änderung auf jeden Fall *vor* dem Aufruf des zugehörigen `\pagestyle{...}`- bzw. `\begin{letter}`-Befehls ausgeführt werden müssen. Als Beispiel sollen die Vorbelegungen der Styledatei dienen[4]:

```
\firsthead{\parbox[b]{\textwidth}{\begin{center}
            {\sc \ignorespaces \fromname}\\[-8pt]
            \rule{\textwidth}{.4pt}\\
            \ignorespaces \fromaddress \end{center}}}
\firstfoot{}
\nexthead{\parbox[b]{\textwidth}{{%
            \sc \ignorespaces \fromname}\\[8pt] \headtoname\
            \ignorespaces \toname \hfill \datename:\ \today
            \hfill \pagename\ \thepage\\[-8pt]
            \rule{\textwidth}{.4pt}}}
\nextfoot{}
```

[4]Das Verständnis der Definitionen ist zugegebenermaßen nicht ganz einfach. Die Bedeutung der verwendeten Variablen schlagen Sie bitte in der Variablenübersicht in Abschnitt 4.3 nach.

Dadurch ist es nun möglich, eigene Briefköpfe und Fußleisten (z.B. für Bankverbindungen, Gerichtsstand etc.) zu entwerfen. Der Phantasie sind eigentlich nur durch die zur Verfügung stehenden Zeichensätze Grenzen gesetzt; Ausgefallenes läßt sich durch das `picture`-Environment oder „Metafont“ realisieren.

4.2.8 Adreßdateien

Um nicht in jedem Brief die komplette Anschrift des Adressaten nachschlagen und vor allem eingeben zu müssen, wurden Adreßdateien (`.adr`-Files) eingeführt. Sie enthalten für jeden Adressaten einen Eintrag in Form eines `\adrentry`-Befehles, der acht Argumente mit folgender Bedeutung hat:

1. Argument: Name

2. Argument: Vorname

3. Argument: Adresse

4. Argument: Telefonnummer

5. Argument: zur freien Verfügung

6. Argument: zur freien Verfügung

7. Argument: Kommentar

8. Argument: Kürzel

Eine solche Adreßdatei bindet man mit `\input` entweder direkt in den Quelltext des Briefes (am besten noch in der Präambel) oder in das oben vorgeschlagene Optionsfile, das die Voreinstellungen enthält, ein.

Beispiel: Mein persönliches Optionsfile sieht folgendermaßen aus:

```
% privat.sty
\name{Frank Neukam}
\address{Franz-Stadelmayer-Str. 30\\97074 W"urzburg}
\backaddress{F. Neukam, F.-Stadelmayer-Str. 30, 97074 W"urzburg}
\place{W"urzburg, }
\input{privat.adr}
```

Für jeden `.adr`-Eintrag wird dann die komplette Anschrift unter dem jeweiligen Kürzel als TeX-Makro abgelegt. Um das Überschreiben von TeX- oder LaTeX-Befehlen zu vermeiden, ist es empfehlenswert, für Kürzel nur Großbuchstaben zu verwenden. Lautet also z.B. ein `.adr`-Eintrag

```
\adrentry{abc-Lernsysteme}
         {}
         {z.Hd. Frau Streb\\Schulstr.1\\2345 Paukstadt}
         {0\,678 / 90\,12\,34}
         {}
         {}
         {}
         {ABC}
```

so schreibt man anstelle der kompletten Adresse nur noch

```
\begin{letter}{\ABC}
```

Mit Hilfe des im `script`-Paket enthaltenen `dir.tex`-Files kann aus einer `.adr`-Datei ein Adreßheft erzeugt werden, das nach Zurechtschneidung in die Standard-Taschenkalenderhüllen paßt. Dazu sollten die Einträge alphabetisch sortiert sein und vor jedem neu begonnenen Anfangsbuchstaben des Nachnamens ein

`\adrchar{`*Anfangsbuchstabe*`}`

stehen. Vom `script_l.sty` wird dieser Befehl ignoriert; `dir.tex` jedoch beginnt bei einer solchen Anweisung eine neue Seite und schreibt diesen Buchstaben in die Kopfzeile. Am besten probieren Sie dieses Programm einfach einmal anhand einer kleinen Adreßdatei aus, alle benötigten Informationen werden während des TeX-Kompilerlaufes erfragt.

Ähnlich wie `dir.tex` arbeitet das Programm `phone.tex`, nur daß dieses anstelle eines Adreßheftes ein Telefonnummern-Verzeichnis anlegt. `\adrchar` erzeugt hier lediglich einen kleinen vertikalen Zwischenraum.

4.2.9 Serienbriefe

Eine weitere Anwendung der `.adr`-Dateien sind Serienbriefe.

Beispiel: Mit der folgenden kurzen Datei kann eine Einladung an sämtliche Kollegen eines Betriebes erzeugt werden:

```
\documentstyle[privat,german]{script_l}
\begin{document}
\def\adrentry#1#2#3#4#5#6#7#8{
 \begin{letter}{#2 #1\\#3}
 \opening{Liebe Kolleginnen und Kollegen,}
 anl"a"slich meines ...
 \closing{Mit freundlichen Gr"u"sen}
 \end{letter}
}
\input{kollegen.adr}
\end{document}
```

Soll der Brieftext abhängig von Adressatenmerkmalen, die in den beiden noch freien Argumenten des `\adrentry`-Befehls stehen, veränderlich sein, können Konstrukte der folgenden Form verwendet werden[5]:

```
\def\adrentry#1#2#3#4#5#6#7#8{
 \ifcase #6
  % Dieser Zweig wird bei #6 = 0 abgearbeitet.
  % In diesem Beispiel sind Empf"anger ohne Kinder uninteressant,
  % weshalb im \else-Zweig nur Briefe an Kunden mit mindestens
  % einem Kind erzeugt werden.
 \else
  % #6 > 0
  \begin{letter}{#2 #1\\#3}
   \if #5m \opening{Sehr geehrter Herr #1,} \fi
   \if #5w \opening{Sehr geehrte Frau #1,} \fi

   Sie als% Direkt nach 'als' mu"s ein % stehen, da der \if-Befehl
          % vor 'gl"ucklich..' einen zus"atzlichen Leerraum
          % generiert.
   \if #5m gl"ucklicher Vater \fi
   \if #5w gl"uckliche Mutter \fi
   wissen sicher am besten, was gut f"ur
   \ifnum #6 < 2
    Ihr Kind
   \else
```

[5] Als Beispiel ist das Geschlecht des Empfängers im Argument #5 kodiert : m für männlich, w für weiblich. In Argument #6 steht die Anzahl der Kinder.

```
      Ihre Kinder
     \fi
     ist. Um Sie...

     \closing{Mit freundlichen Gr"u"sen}
    \end{letter}
   \fi
  }
```

Es ist also auch möglich, den Brieftext auf bestimmte Empfängermerkmale gezielt abzustimmen und so den Eindruck eines persönlichen Schreibens zu erwecken. Die Anwendungsbreite ist lediglich durch die maximale Anzahl von zwei freien `\adrchar`-Argumenten begrenzt.

4.3 Befehls- und Variablenübersicht

Briefspezifische Befehle, die strukturbeschreibend sind oder eine Ausgabe erzeugen:

`\begin{letter}{`*Adressat*`}`	Markiert den Beginn eines Briefes an *Adressat* und beginnt eine neue Seite
`\end{letter}`	Markiert das Ende eines Briefes
`\opening{`*Anrede*`}`	Setzt alle Teile eines Briefes oberhalb und einschließlich der *Anrede*
`\closing{`*Grußformel*`}`	Setzt *Grußformel* und Unterschrift
`\ps{`*Postskriptum*`}`	Setzt ein *Postskriptum*
`\cc{`*Verteiler*`}`	Setzt eine Verteilerliste (Einträge durch `\\` abgetrennt)
`\encl{`*Anlagen*`}`	Setzt eine Anlagenliste (Einträge durch `\\` abgetrennt)

Befehle der Adreßdateien:

`\adrchar` und `\adrentry` (s.o.)

Befehle zur Sprachumschaltung:

`\captionsenglish`	Selektiert englische Caption-Texte
`\captionsUSenglish`	Selektiert amerikanische Caption-Texte
`\captionsgerman`	Selektiert deutsche Caption-Texte
`\captionsaustrian`	Selektiert österreichische Caption-Texte
`\captionsfrench`	Selektiert französische Caption-Texte
`\dateenglish`	Englisches Datum (1/12/1993)
`\dateUSenglish`	Amerikanisches Datum (12/1/1993)

`\dategerman`	Deutsches Datum (1. 12. 1993)
`\dateaustrian`	Österreichisches Datum (1. 12. 1993)
`\datefrench`	Französisches Datum (1. 12. 1993)

Sprachabhängige Variablen

(dürfen an beliebiger Stelle nach dem `\begin{document}`-Befehl verwendet werden, können aber nur mit Hilfe von `\def` oder `\renewcommand` geändert werden; die Liste enthält die Variableninhalte nur in englisch, deutsch und französisch, da die amerikanischen Namen den englischen und die österreichischen den deutschen entsprechen):

`\yourrefname`	Your ref. / Ihr Zeichen / Vos références
`\yourmailname`	Your letter of / Ihr Schreiben vom / Votre lettre du
`\myrefname`	Our ref. / Unser Zeichen / Nos références
`\customername`	Customer no. / Kundennummer / Numéro de client
`\invoicename`	Invoice no./ Rechnungsnummer / Numéro de facture
`\subjectname`	Subject / Betr. / Concernant
`\ccname`	cc / Kopien an / Copie à
`\enclname`	encl / Anlagen / Annexes
`\headtoname`	To / An / A
`\datename`	Date / Datum / Date
`\pagename`	Page / Seite / Page

Briefspezifische Variablen und deren Befehle zur Neu- oder Umdefininierung

(Variablen mit `@` sind geschützt; alle anderen können an beliebiger Stelle verwendet werden):

`\name`	ändert `\fromname`

`\signature`	ändert	`\fromsig`
`\address`	ändert	`\fromaddress`
`\place`	ändert	`\fromplace`
`\location`	ändert	`\fromlocation`
`\backaddress`	ändert	`\frombackaddress`
`\telephone`	ändert	`\telephonenum`
`\yourref`	ändert	`\varyourref`
`\yourmail`	ändert	`\varyourmail`
`\myref`	ändert	`\varmyref`
`\customer`	ändert	`\varcustomer`
`\invoice`	ändert	`\varinvoice`
`\refitemi`	ändert	`\varrefitemi`
`\refitemii`	ändert	`\varrefitemii`
`\refitemiii`	ändert	`\varrefitemiii`
`\refitemnamei`	ändert	`\varrefitemnamei`
`\refitemnameii`	ändert	`\varrefitemnameii`
`\refitemnameiii`	ändert	`\varrefitemnameiii`
`\specialmail`	ändert	`\@specialmail`
`\title`	ändert	`\@title`
`\subject`	ändert	`\@subject`
`\firsthead`	ändert	`\@firsthead`
`\firstfoot`	ändert	`\@firstfoot`
`\nexthead`	ändert	`\@nexthead`
`\nextfoot`	ändert	`\@nextfoot`

Briefspezifische Längenangaben

(voreingestellte Länge in eckigen Klammern; die Maße `\foldvskipi` bis `\foldvskipiii` sind erst ab der Version 2.0 Bestandteil des `script_l.sty`):

`\foldhskip`	Abstand der Falzmarken vom linken Papierrand [3.5 mm]
`\foldvskipi`	Abstand der ersten Falzmarke vom oberen Papierrand [65 mm]
`\foldvskipii`	Abstand der zweiten Falzmarke von der ersten Falzmarke [40.5 mm]
`\foldvskipiii`	Abstand der dritten Falzmarke von der zweiten Falzmarke [58.5 mm]
`\addrvskip`	Abstand des Adreßfensters von der Textbereichs-Oberkante [7.5 mm]
`\addrindent`	Abstand des Adreßfensters vom linken Rand des Textbereiches [0 mm]
`\addrwidth`	Breite des Adreßfensters [70 mm]
`\addrheight`	Höhe des Adreßfensters [35 mm]
`\locwidth`	Breite des „Location"-Feldes [(`\textwidth` − `\addrwidth`)/2]
`\refvskip`	Abstand des Referenzfeldes von der Adreßfeld-Unterkante [5.5 mm]
`\sigindent`	Abstand der Grußformel und der Unterschrift vom linken Rand des Textbereiches [0 mm]

Schalter

(an jeden Schalternamen ist noch `on` bzw. `off` anzuhängen):

`\foldmarks`	schaltet Falzmarken [Default = on]
`\addrfield`	schaltet Adreß- und „Location"-Feld [Default = on]
`\subject`	schaltet „Betreff"-Text [Default = off]