

Die Document-Style-Familie „Script“

Version 1.0

6. Juni 1992

Haftungsausschluß:

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

Inhaltsverzeichnis

1. Allgemeines	3
1.1 Einleitung	3
1.2 Autor, Copyright und Danksagungen	3
1.3 Was ist was	4
1.4 Kompatibilität	4
1.5 Fehler	5
2. Das neue Layout	6
2.1 Die neuen Fonts	6
2.2 Der Satzspiegel	6
2.2.1 Grundlegendes	6
2.2.2 Konstruktion	7
2.2.3 Auswahl des gewünschten Satzspiegels	7
2.2.4 Ungewöhnliche Satzspiegel	9
2.2.5 Berücksichtigung der Kopfzeile	10
2.2.6 Einbindung der Befehle zur Satzspiegeleinstellung	10
2.2.7 Andere Papierformate	10
2.3 Sonstige Änderungen	11
2.3.1 Default-Fontgröße	11
2.3.2 Kapitelüberschriften und Nummerierung	11
2.3.3 Kolumnentitel und Pagina	11
2.3.4 Kleinere kosmetische Veränderungen	12
3. Zusätzliche Befehle	14
3.1 Die Erweiterungen des Abstracts und der Titelseite	14
3.2 Die „marginline“	15
3.3 Der „addchap“- bzw. „addsec“-Befehl	15
3.4 Der „minisec“-Befehl	16
3.5 Das „labeling“-Environment	16

4. Der script_I-Style	18
4.1 Das Original	18
4.2 Zusätzliche Eigenschaften des script_I-Styles	19
4.2.1 Dreisprachigkeit	19
4.2.2 Zweiseitiger Druck	19
4.2.3 Layout	20
4.2.4 Adreßfeld und Zusatz-„location“-Feld	20
4.2.5 Referenzfeld	21
4.2.6 Titel und Betreff	21
4.2.7 Seitenstile	22
4.2.8 Adreßdateien	22
4.2.9 Serienbriefe	24
4.3 Befehls- und Variablenübersicht	25

1. Allgemeines

1.1 Einleitung

Ursprünglich versuchte ich im Rahmen eines größeren Projektes¹ 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 Typographie² 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. Einen ersten Eindruck von den daraufhin durchgeführten Modifikationen kann man durch diese Dokumentation selbst bekommen; sie wurde mit `\documentstyle[twoside,german]{script}` und nachträglicher Einstellung von `\headseplineon` (siehe S. 12) gesetzt. In den folgenden Kapiteln wird jede Neuerung ausführlich beschrieben, wobei allerdings gewisse Grundkenntnisse im Umgang mit \LaTeX erforderlich sind.

1.2 Autor, Copyright und Danksagungen

Die Document-Style-Familie `script` und die dazugehörige Dokumentation (die Sie gerade lesen) wurden von Frank Neukam geschrieben. Nachträgliche Änderungen 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 Kommentarseiten wie „Modified by ...“ einzufügen. Außerdem sollte dann in der typeout message und der ersten Kommentarseite das Datum aktualisiert und jede Änderung kommentiert werden. Ansonsten sind alle „Mitglieder“ der `script`-Familie natürlich *public domain*!

Mein besonderer Dank gilt an dieser Stelle dem Rechenzentrum der Universität Karlsruhe (für viele Probeausdrucke) sowie Thorsten Hoeppner, Uta Janssen, Klaus-Jürgen Wald und Christian Wurl, ohne deren Ideen, Mithilfe und Kritik dieses Programmpaket

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

²JAN TSCHICHOLD: Ausgewählte Aufsätze über die Gestalt des Buches und der Typographie, Zweite Auflage, Birkhäuser Verlag Basel 1987

wohl nicht in dieser Form zustande gekommen wäre.

1.3 Was ist was

Das komplette Paket besteht aus folgenden Files:

<code>checkprt.tex</code>	Testseite zur genauen Einstellung von Druckertreibern
<code>dir.tex</code>	Programm zum Drucken eines Adreßbuches aus einem <code>.adr</code> -File (siehe S. 22)
<code>script.dvi</code>	Dokumentation zur „Script“-Stylefamilie
<code>script.sty</code>	Style-Hauptdatei; entspricht einem Mittelding aus <code>report.sty</code> und <code>book.sty</code>
<code>script.tex</code>	Quelltext zu <code>script.dvi</code>
<code>script10.sty</code>	Hilfsdatei für alle drei Hauptstyles; wird von diesen bei Einstellung der 10pt-Schrift nachgeladen
<code>script11.sty</code>	Dito, für 11 pt
<code>script12.sty</code>	Dito, für 12 pt
<code>script_l.sty</code>	Style-Hauptdatei „script_letter“; entspricht dem <code>letter.sty</code>
<code>script_s.sty</code>	Style-Hauptdatei „script_small“; entspricht dem <code>article.sty</code>
<code>sctitle.sty</code>	Pendant zu <code>titlepage.sty</code> ; erzeugt eine eigene Titelseite
<code>typearea.sty</code>	Style-Option zur einfacheren Satzspiegeleinstellung

1.4 Kompatibilität

Alle Befehle, die in den zugrundeliegenden Original-Styles vorhanden sind, sind auch weiterhin verfügbar. Es wurden lediglich Befehle hinzugefügt bzw. schon vorhandene Einstellungen und Befehlsdefinitionen modifiziert. Dadurch muß man bei Verwendung der neuen 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 bei einem evtl. 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
8700 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 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` (analog zu `\bf`) 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 Seite¹ 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 mit dem Seitenverhältnis $1:\sqrt{2}$, von dem in allen `script`-Hauptstyles 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).

¹Im folgenden auch „Textbereich“ genannt; Kopfzeile, Fußzeile und Randnoten zählen jedoch nicht dazu.

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 $\text{HLE} = 210 \text{ mm} \div 10 = 21 \text{ mm}$ und für die vertikale Längeneinheit $\text{VLE} = 297 \text{ mm} \div 10 = 29.7 \text{ mm}$.

Nun wird bei doppelseitigem Druck einfach der innere Rand² 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 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 (Randbemerkun-

²Es 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.

Textfontgröße	<i>DIV</i>
10pt	8
11pt	10
12pt	12

Tabelle 2.1: *DIV*-Voreinstellungen

gen 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* schmaler 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 Text Quelldatei vor `\begin{document}`) 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.

Fehlt der `\typearea`-Befehl in der Präambel, so wird *DIV* nach Tabelle 2.1 gewählt, wobei keine Bindungs-Korrektur vorgenommen wird (*BCOR* = 0).

Als Anhaltspunkt für die manuelle *DIV*-Einstellung kann Tabelle 2.2 dienen. Dabei wurde *BCOR* = 0 und das Papierformat DIN A4 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.

<i>DIV</i>	Textbereich		Ränder	
	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*

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} % Textbereichshöhe: 20cm
```

Auch dieser Befehl darf selbstverständlich nur in der Präambel auftreten.

2.2.5 Berücksichtigung der Kopfzeile

Manchmal ist es sinnvoll, daß bei der Konstruktion des Satzspiegels auch die Kopfzeile 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.

2.2.6 Einbindung der Befehle zur Satzspiegeleinstellung

In die beiden Hauptstyles `script` und `script_s` sind die beschriebenen Befehle 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 separat 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).

2.2.7 Andere Papierformate

Sollen sich die Satzspiegelberechnungen nicht auf DIN A4 (wie es hier voreingestellt ist), sondern auf ein anderes Papierformat beziehen, so müssen `\ta@pw` und `\ta@ph` in `typearea.sty` verändert werden. Dabei ist `\ta@pw` die Papierbreite (*paperwidth*; DIN A4: 210 mm) und `\ta@ph` die Papierhöhe (*paperheight*; DIN A4: 297 mm).

Beispiel: Ein Gedichtbändchen soll im DIN A5-Format gedruckt werden. Dazu erstellt man eine Kopie von `typearea.sty` und benennt diese in `a5lyrics.sty` um. Daraufhin sucht man nach den Zeilen

```
\ta@pw 210mm  
\ta@ph 297mm
```

und ändert diese zu

```
\ta@pw 148.5mm  
\ta@ph 210mm
```

Schreibt man nun im Quelltext dieses Gedichtbändchens z.B.

```
\documentstyle[10pt,german,a5lyrics]{book}
```

so wird der Satzspiegel automatisch auf DIN A5 berechnet.

Die so neu erstellte Style-Option läßt sich selbstverständlich auch bei `script.sty` und `script_s.sty` verwenden; die dort schon definierten Größen und Befehle zur Satzspiegelberechnung werden dann durch das Options-File einfach überschrieben.

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_1`-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), 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.

2.3.4 Kleinere kosmetische Veränderungen

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

In Fußnoten ist nun 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.

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 jetzt, wie eigentlich auch hierzulande üblich, a) statt (a) geschrieben wird. Weiterhin wurde in der dritten Ebene des `itemize`-Environments der „*“ durch ein „▷“ ersetzt (das sieht zumindest für meinen Geschmack etwas gefälliger aus).

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

In mehrzeiligen Captions (Unterschriften von `table` bzw. `figure`) werden nun 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 vergrößert 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 Haftungsausschluß) nutzen. Dazu stehen die Befehle `\uppertitleback` und `\lowertitleback` zur Verfügung.
- `\dedication{Widmung}` erzeugt eine separate Widmungsseite, wobei *Widmung* zentriert und in etwas größerer Schrift gesetzt wird. Wie bei `\extratitle` ist die Rückseite grundsätzlich leer.

¹Früher 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.

Beispiel:

```
\titlehead{Institut f"ur Nachrichtentechnik \hfill SS1992
           \\Hochschulstr.12
           \\3456 Unistadt}
\subject{Diplomarbeit}
\title{Digitale Raumsimulation mit dem MC96000}
\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 `scitle.sty` geschrieben.

3.2 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.

3.3 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 L^AT_EX-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 Kolummentitel gelöscht werden (geht z.B. der zu einem `\chapter*` gehörende Text über mehrere Seiten, so tauchen die – veralteten! – Kolummentitel plötzlich wieder auf). Zusätzlich ist als Optionalargument wie bei den Originalen ein anderer Eintrag ins Inhaltsverzeichnis als die Abschnittsüberschrift angebar.

3.4 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_1`-Style.

3.5 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_I-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).

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 23.

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 Dreisprachigkeit

Der *script_l*-Style wurde von vorneherein dreisprachig ausgelegt (englisch, deutsch, französisch; Default ist englisch). Die Wahl der Sprache erfolgt wie gehabt durch die `\documentstyle`-Option `german` bzw. `french`. Für `french` ist im Gegensatz zu `german` keine eigene Style-Datei `french.sty` erforderlich.

Durch die Sprachumschaltung wurde erreicht, daß automatisch gesetzte Texte (wie „Betreff“, „Seite“, „Anlagen“ usw.) in der gewünschten Sprache erscheinen. Auch das Datum wird in der landesspezifischen Kurzform geschrieben:

englisch : 12/1/1992

deutsch : 1.12.1992

französisch: 1.12.1992

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 *Druckseite* beginnt. Die *interne* T_EX-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.

Kopf- und Fußbereich der Seite sind 20 mm hoch; sie sind vom oberen bzw. unteren Rand und vom Textbereich jeweils 10 mm entfernt. Linker und rechter Rand sind beide 25.4 mm breit. Dadurch bleibt für den Textbereich eine Fläche von $(297\text{ mm} - 2 \cdot 20\text{ mm} - 4 \cdot 10\text{ mm}) \times (210\text{ mm} - 2 \cdot 25.4\text{ mm}) = 217\text{ mm} \times 159.2\text{ mm}$.

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`).

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.

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 „address“ 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{Kurzzinhalt 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 eigentlich Clou an der Sache ist nun, daß die Einträge in den Kopf- und Fußbereich der Seiten bei `firstpage` und `headings` frei definierbar sind. Dies geschieht durch die Befehle `\firsthead`, `\firstfoot`, `\nexthead` und `\nextfoot`, was am Beispiel der Vorbelegungen der Styledatei vorgeführt werden soll¹:

```
\firsthead{\parbox[b]{\textwidth}{\begin{center}\xiipt
\baselineskip 14.5pt {\sc \ignorespaces \fromname}
\\[-8pt] \rule{\textwidth}{.4pt}\\
\ignorespaces \fromaddress \end{center}}}\n
\firstfoot{}
\nexthead{\parbox[b]{\textwidth}{\xiipt \baselineskip 14.5pt
\sc \ignorespaces \fromname}\\[8pt] \headtoname\
\ignorespaces \toname \hfill \datename:\ \today
\hfill \pagename\ \thepage\\[-8pt]
\rule{\textwidth}{.4pt}}}\n
\nextfoot{}
```

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

¹Das Verständnis der Definitionen ist zugegebenermaßen nicht ganz einfach; als kleine Erläuterung mag dienen, daß `\xiipt` auf 12pt-Schrift umschaltet und `\ignorespaces` alle nachfolgenden Leerräume bis zum ersten anderen Zeichen ignoriert. Die Bedeutung der verwendeten Variablen schlagen Sie bitte in der Variablenübersicht in Abschnitt 4.3 nach.

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\\8700 W"urzburg}
\backaddress{F. Neukam, F.-Stadelmayer-Str. 30, 8700 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.

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 Briefftext abhängig von Adressatenmerkmalen, die in den beiden noch freien Argumenten des `\adrentry`-Befehls stehen, veränderlich sein, können Konstrukte der folgenden Form verwendet werden²:

```
\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
```

²Als Beispiel ist das Geschlecht des Empfängers im Argument `#5` kodiert : m für männlich, w für weiblich. In Argument `#6` steht die Anzahl der Kinder.

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

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

Es ist also auch möglich, den Briefftext 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:

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

Befehle der Adreßdateien:

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

Sprachabhängige Variablen

(dürfen an beliebiger Stelle verwendet werden, können aber nur mit Hilfe von `\def` oder `\renewcommand` geändert werden):

`\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 Umdefinierung

(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`

<code>\telephone</code>	ändert <code>\telephonenumber</code>
<code>\yourref</code>	ändert <code>\varyourref</code>
<code>\yourmail</code>	ändert <code>\varyourmail</code>
<code>\myref</code>	ändert <code>\varmyref</code>
<code>\customer</code>	ändert <code>\varcustomer</code>
<code>\invoice</code>	ändert <code>\varinvoice</code>
<code>\refitemi</code>	ändert <code>\varrefitemi</code>
<code>\refitemii</code>	ändert <code>\varrefitemii</code>
<code>\refitemiii</code>	ändert <code>\varrefitemiii</code>
<code>\refitemnamei</code>	ändert <code>\varrefitemnamei</code>
<code>\refitemnameii</code>	ändert <code>\varrefitemnameii</code>
<code>\refitemnameiii</code>	ändert <code>\varrefitemnameiii</code>
<code>\specialmail</code>	ändert <code>\@specialmail</code>
<code>\title</code>	ändert <code>\@title</code>
<code>\subject</code>	ändert <code>\@subject</code>
<code>\firsthead</code>	ändert <code>\@firsthead</code>
<code>\firstfoot</code>	ändert <code>\@firstfoot</code>
<code>\nexthead</code>	ändert <code>\@nexthead</code>
<code>\nextfoot</code>	ändert <code>\@nextfoot</code>

Briefspezifische Längenangaben

(voreingestellte Länge in eckigen Klammern):

<code>\foldhskip</code>	Abstand der Falzmarken vom Papierrand [3.5 mm]
<code>\addrvskip</code>	Abstand des Adreßfensters von der Textbereichs-Oberkante [7.5 mm]
<code>\addrindent</code>	Abstand des Adreßfensters vom linken Rand des Textbereiches [0 mm]
<code>\addrwidth</code>	Breite des Adreßfensters [70 mm]

<code>\addrheight</code>	Höhe des Adreßfensters [35 mm]
<code>\locwidth</code>	Breite des „Location“-Feldes $(\text{\textwidth} - \text{\addrwidth})/2$
<code>\refvskip</code>	Abstand des Referenzfeldes von der Adreßfeld-Unterkante [5.5 mm]
<code>\sigindent</code>	Abstand der Grußformel und der Unterschrift vom linken Rand des Textbereiches [0 mm]

Schalter

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

<code>\foldmarks</code>	schaltet Falzmarken [Default = on]
<code>\addrfield</code>	schaltet Adreß- und „Location“-Feld [Default = on]
<code>\subject</code>	schaltet „Betreff“-Text [Default = off]