KOMA-Script – The Source*

Markus Kohm

2011-02-19

Version 3.08a

In diesem Dokument finden Sie *nicht* die Anleitung zu KOMA-Script. Wenn Sie diese suchen, dann suchen Sie bitte nach `scrguide` (Deutsch) oder `scrguien` (Englisch). In diesem Dokument ist die Implementierung von KOMA-Script – in erster Linie der KOMA-Script-Klassen – dokumentiert.

Inhaltsverzeichnis

Vorwort **9**

a scrkvers.dtx **10**

1 Version des KOMA-Script-Kerns **10**

b scrkbase.dtx **12**

2 Grundlegenden KOMA-Script-Erweiterungen **12**

2.1 Laden des Pakets 12
2.2 ε-TEX-Test 12
2.3 Der abhängige Teil lädt den unabhängigen Teil 13
2.4 Laden von keyval 13
2.5 Test von Anweisungen 13
2.6 Definition von KOMA-keyval-Schlüsseln (Basis) 13
2.7 Optionen 15
2.8 Ein wenig zusätzliche Mathematik für ε-TEX 18

*Ich weiß natürlich, dass ein Englischer Titel eines vorwiegend deutschsprachigen Dokuments ein wenig eigentümlich ist, aber "'KOMA-Script – Der Quelltext"' wollte mir einfach nicht gefallen.

2.9 Häufig benötigte Befehle 18
2.10 Definition von KOMA-keyval-Schlüsseln (Hauptteil) 25

c scrkmisc.dtx 39

3 Verschiedenes 39
3.1 Erweiterung der Internationalisierung 39
3.2 Optionen für Entwürfe 39
3.3 Optionen für Gleichungen 40
3.4 Standardeinstellungen für die Optionen 40
3.5 Temporäre Umschaltung zwischen ein- und zweispaltigem Satz 40
3.6 Definitionen für den schlauen Spruch 41
3.7 Längen für Minipages und Rahmen 42
3.8 Gleichungen 42
3.9 Definitionen zur Feststellung von linken und rechten Seiten 42
3.10 Definitionen für tiefer gestellten Text 44

d scrklang.dtx 45

4 Sprachabhängigkeiten 45
4.1 Option 45
4.2 Definitionen für sprachabhängige Bezeichner 45
4.3 Definitionen für ein nummerisches Datum 48
4.4 Festlegungen für einzelne Sprachen 48

e scrkfont.dtx 59

5 Schriftauswahl 59
5.1 Option zur Auswahl der Größe der Grundschrift 59
5.2 Einlesen der Schriftgrößendatei 60
5.3 „Alte“ Font-Auswahlbefehle 72
5.4 Setzen der Schrift eines Elements 72

f scrktare.dtx 76

6 Satzspiegel Makros 76
6.1 Präambel 76
6.2 Von den Optionen benötigte Befehle 76
6.3 Optionen mit keyval 81
6.3.1 Kopf-, Fuß-, Rand- und Textbereich 81
6.3.2 Sonstige Satzspiegel-Optionen 83

6.3.3 Ausgabeformat-Optionen 84
6.3.4 Papiergrößen-Optionen 88
6.3.5 Geparste Optionen 91
6.3.6 Konfigurationsdatei 91
6.3.7 Standardoptionen 92
6.4 Schriftgrößenabhängige Voreinstellungen für *DIV* 92
6.5 Laden des Pakets 92
6.6 Einige interne Register 93
6.7 Die Berechnung des Satzspiegels 93
6.8 Berechnung des Standardsatzspiegels 107

g scrkfloa.dtx 109

7 Gleitumgebungen 109
7.1 Einige Anweisungen und Einstellungen für die Optionen . . 109
7.2 Optionen für Titel von Gleitumgebungen 110
7.3 Randnotizen 111
7.4 Definitionen für Titel von Gleitumgebungen 111
7.5 Verteilung der Gleitumgebungen 125
7.6 Titel ohne Gleitumgebung 125
7.7 Standardbezeichner 125
7.8 Abbildungsumgebung 126
7.9 Tabellengleitumgebung 127

h scrkftn.dtx 129

8 Fußnoten 129
8.1 Option 129
8.2 Definitionen für Fußnoten 130
8.3 Fußnotenreferenz 134

i scrkpage.dtx 135

9 Seitenstil 135
9.1 Optionen für die Trennlinien in Kopf und Fuß 135
9.2 Optionen und Anweisungen für leere Seiten 136
9.3 Befehle für den Kompatibilitätsmodus 137
9.4 Definition der Standardseitenstile 137
9.5 Festlegung des Seitenstils auf besonderen Seiten 142
9.6 Standardeinstellungen 142

j scrkpar.dtx **144**

10 Absatzformatierung und -umbruch **144**
10.1 Optionen für das Absatzlayout 144
10.2 Abatzformatierung 147
10.3 Umbruchsteuerung 151

k scrktitl.dtx **152**

11 Die Titelei **152**
11.1 Optionen der Titelei 152
11.2 Definitionen der Titelei 153
11.3 Umgebung für die Titelseite 158
11.4 Fonts für den Titel 158
11.5 Umgebung für die Zusammenfassung 158

l scrksect.dtx **160**

12 Gliederung **160**
12.1 Optionen für die Gliederung 160
12.2 Grobeinteilung von Büchern 166
12.3 Anhang 166
12.4 Definitionen für Seitenstile 167
12.5 Der Automatismus für den Punkt in der Nummerierung . . 167
12.6 Zähler für die Gliederungsebenen 168
12.7 Namen für die Gliederung 170
12.8 Die Gliederungsbefehle 171
12.9 Schrift und Formatierung 183
12.10 Definitionen für Präambeln 184
12.11 Einträge ins Inhaltsverzeichnis 187

m scrkliof.dtx **192**

13 Verzeichnisse **192**
13.1 Optionen für das Inhaltsverzeichnis 192
13.2 Optionen für die Gleitumgebungen 193
13.3 Definitionen für alle Verzeichnisse 197
13.4 Definitionen für das Inhaltsverzeichnis 199
13.5 Definitionen für die Verzeichnisse der Gleitumgebungen . . 200

n scrkbib.dtx **203**

14 Literaturverzeichnis **203**
14.1 Anweisungen, die für die Optionen benötigt werden 203
14.1.1 Literaturverzeichnisstile 203
14.1.2 Überschrift des Literaturverzeichnisses 205
14.2 Optionen für das Literaturverzeichnis 205
14.3 Standardbezeichner 206
14.4 Definitionen für das Literaturverzeichnis 206

o scrkidx.dtx **210**

15 Index **210**
15.1 Anweisungen und Einstellungen für die Optionen 210
15.2 Optionen für den Index 210
15.3 Definitionen für den Index 211
15.4 Standardbezeichner 212

p scrklist.dtx **213**

16 Listen und Tabellen **213**
16.1 Optionen 213
16.2 Ränder in Listen 213
16.3 Verschiedene Längen für Tabellen, Arrays und Tabulatoren 214
16.4 Marken in Listen 214
16.5 Definition der Umgebungen 214
16.6 Schriftarten für Listen 217

q scrlfile.dtx **218**

17 Hook loading files, packages or classes **218**
17.1 Preamble 222
17.2 Options 222
17.3 The macros 223

r scrlogo.dtx **231**

18 KOMA-Script Logo **231**

s scrkcomp.dtx **232**

19 Kompatibilität zu früheren Versionen **232**
19.1 Option 232
19.2 Kompatibilität mit früheren Versionen von scrlttr2 234

t scrknpap.dtx **235**

20 Briefbogen und Briefpapier **235**
20.1 Option für den Briefbogen 235
20.1.1 Optionen für den Briefkopf 235
20.1.2 Optionen für Anschrift und Absenderergänzung . . . 237
20.1.3 Optionen für Faltmarken 240
20.1.4 Optionen für die Geschäftszeile 241
20.1.5 Optionen für den Betreff 242
20.1.6 Optionen für den Satzspiegel 243
20.1.7 Optionen für den Briefbogenfuß 243
20.2 Optionen für den Kopf des weiteren Briefpapiers 243
20.3 Anweisungen für den Briefbogen 245
20.3.1 Der Kopf des Briefbogens 245
20.3.2 Schrifteinstellungen für den Briefbogen 249
20.3.3 Längen für den Briefbogen 251
20.3.4 Fuß des Briefbogens 253
20.3.5 Ausgabe des Briefbogens 254
20.4 Anweisungen für das Briefpapier 274
20.4.1 Kopf des Briefpapiers 274
20.4.2 Fuß des Briefpapiers 275
20.5 Die Brief-Umgebung 276
20.6 Briefspezifische Feld-Variablen 280

u scrkvars.dtx **282**

21 KOMA-Script-Variablen **282**
21.1 Option 282
21.2 Anweisungen zum Anlegen und Verwenden von Variablen . 282

v scrkplen.dtx **287**

22 Pseudo-Längen für Briefe **287**
22.1 Option 287
22.2 Anweisungen für Pseudo-Längen 287

w scrklco.dtx **290**

23 Optionendateien für Briefe **290**
23.1 Optionen für Letter-Class-Options 290
23.2 Makros für Letter-Class-Options 290
23.3 Die Letter-Class-Optionen 293
23.3.1 Der Datei Kopf 293
23.3.2 Der Hauptteil 293
23.4 Makros zur Visualisierung diverser Felder in einem Brief . . 299

x japanlco.dtx **313**

24 Option Files for Japanese Letters **313**
24.0.1 The File Header 313
24.0.2 Main-Part 314

y scrkcile.dtx **316**

25 Serienbriefe und Adressdateien **316**
25.1 Option 316
25.2 Makos für Serienbriefe und Adressdateien 316

z scrextend.dtx **317**

26 Erweiterungen für andere Klassen **317**
26.1 Vorbereitung noch vor der Identifizierung des Pakets 317
26.2 Identifizierung des Pakets 317
26.3 Vorbereitung vor den Optionen 317
26.4 Optionen 318

A scrhack.dtx **320**

27 The hyperref hack **321**

28 The float hack **322**

29 The listings hack **322**

30 Implementation of scrhack **323**
30.1 Optionen 323
30.2 Verwendete Anweisungen 323
30.3 Der hyperref-Hack 324

30.4 Der float-Hack 327
30.5 Der listings-Hack 330
30.6 Optionen ausführen 332

Vorwort

Sie finden im Folgenden die Dokumentation der Implementierung von KOMA-Script. Diese kann für Paketautoren von Interesse sein. Bevor ein Paketautor ein Makro umdefiniert oder ein internes Makro verwendet, sollte er jedoch Rücksprache mit dem Autor von KOMA-Script halten, damit so weit wie möglich sichergestellt ist, dass bei zukünftigen Versionen von KOMA-Script die notwendige Kompatibilität erhalten bleibt oder der Paketautor über notwendige Änderungen an den entsprechenden Makros informiert wird.

Teile der Dokumentation sind in Englisch. Der größte Teil ist jedoch in Deutsch.

Der Quelltext ist in logische Gruppen eingeteilt und in dieser Form dokumentiert. Die Reihenfolge in der Dokumentation entspricht jedoch *nicht* der Reihenfolge des Codes in den Klassen und Paketen.

Datei a scrkvers.dtx

1 Version des KOMA-Script-Kerns

Dies ist die erste Datei in allen Dateien, Klassen und Paketen, die zum KOMA-Script-Kern gehören (Ausnahme: `scrlfile`).

`\KOMAScriptVersion` Dieses Makro gibt an, zu welcher KOMA-Script-Version die Datei gehört. Die Klassen und Pakete des KOMA-Script-Kerns verwenden dieses Makro außerdem als eigene Versionsangabe.

`\@CheckKOMAScriptVersion` Je nachdem, ob `\KOMAScriptVersion` bereits definiert ist nicht, wird die Definition überprüft oder eine globale Definition vorgenommen. Da das ganze auch bei der Erstellung der Doku geschieht, wird mit `\makeatletter` innerhalb einer Gruppe gearbeitet.

```
\begingroup
  \makeatletter
  \ifx\KOMAScriptVersion\undefined
    \newcommand*{\@CheckKOMAScriptVersion}[1]{%
      \gdef\KOMAScriptVersion{#1}%
    }%
  \else
    \newcommand*{\@CheckKOMAScriptVersion}[1]{%
      \def\@tempa{#1}%
      \ifx\KOMAScriptVersion\@tempa\else
        \@latex@warning@no@line{%
          \noexpand\KOMAScriptVersion\space is
          `\KOMAScriptVersion',\MessageBreak
          but `#1' was expected!\MessageBreak
          You should not use classes, packages or files
          from\MessageBreak
          different KOMA-Script versions%
        }%
      \fi
    }
  \fi
  \@CheckKOMAScriptVersion{2011/02/19 v3.08a KOMA-Script}%
\endgroup
```

¡*class¿

`\KOMAClassName` `\ClassName` Für KOMA-Script-Klassen wird hier einmalig der Name der Klasse definiert und und die Klasse der Klasse angegeben.

```
\newcommand*{\KOMAClassName}{%
⟨article⟩  scrartcl%
```

```
⟨report⟩  scrreprt%
⟨book⟩  scrbook%
⟨letter⟩  scrlttr2%
}
\newcommand*{\ClassName}{%
⟨article⟩  article%
⟨report⟩  report%
⟨book⟩  book%
⟨letter⟩  letter%
}
```

```
\ProvidesClass{\KOMAClassName}[%
  \KOMAScriptVersion\space document class (\ClassName)%
]
```

¡/class¿

Datei b
scrkbase.dtx

2 Grundlegenden KOMA-Script-Erweiterungen

Neue Versionen von KOMA-Script machen zunehmend Gebrauch vom keyval-Paket. Das keyval-Paket ist Bestandteil des graphics-Pakets und muss als solches auf allen TEX-Systemen mit LATEX enthalten sein. Es genügt jedoch in einigen Punkten nicht den Anforderungen von KOMA-Script. Deshalb werden hier Erweiterungen vorgenommen, die von KOMA-Script-Klassen und KOMA-Script-Paketen verwendet werden können.

Darüber hinaus werden immer mehr Teile von KOMA-Script von mehr als einem Paket oder von Paketen und Klassen verwendet. Diese grundlegenden Dinge wurden daher in zwei Pakete, scrbase (für Dinge, die auch von anderen Klassen und Paketen genutzt werden können) und scrkbase (für Dinge, die KOMA-Scriptvorbehalten bleiben), ausgegliedert. Dadurch werden die Pakete und Klassen selbst kleiner und die können trotzdem unabhängig voneinander verwendet werden. Selbstverständlich kann man dieses Paket auch ohne den Rest von KOMA-Script zum Einsatz bringen.

2.1 Laden des Pakets

Wer das Paket lädt, tut das allerdings in der Regel sogar bereits vor der Definition der Optionen mit:

```
\RequirePackage{scrkbase}[\KOMAScriptVersion]
```

2.2 ε-TEX-Test

```
⟨*base⟩
\begingroup
  \def\@tempb{}%
  \def\@tempa{%
    \PackageError{scrbase}{seems you are not running e-TeX\@tempb}{%
      Since 2004 the LaTeX team recommends to use e-TeX.\MessageBreak
      KOMA-Script since version 2.95 uses several e-TeX features.\MessageBreak
      At actual systems ‘latex’ should already use e-TeX.\MessageBreak
      At deprecated systems it may be called ‘elatex’.\MessageBreak
      Use either unsupported KOMA-Script up to version 2.9u or\MessageBreak
      ask you administrator for LaTeX using e-TeX\@tempb.\MessageBreak
      Not using e-TeX\@tempb\space is a fatal error!\MessageBreak
      Processing cannot be continued!}%
    \endgroup
    \batchmode \errmessage{}\csname @@end\endcsname\end\relax
    \csname endinput\endcsname
```

```
  }%
  \expandafter\ifx\csname eTeXversion\endcsname\relax\else
    \ifnum\eTeXversion <2
      \def\@tempb{ V 2}%
    \else
      \let\@tempa\endgroup
    \fi
  \fi
\@tempa
⟨/base⟩
```

2.3 Der abhängige Teil lädt den unabhängigen Teil

```
⟨koma⟩\RequirePackage{scrbase}[\KOMAScriptVersion]
```

2.4 Laden von keyval

Hier wird nicht nur das Paket geladen, was ruckzuck mit:

```
⟨base⟩\RequirePackage{keyval}
```

erledigt ist.

2.5 Test von Anweisungen

`\scr@ifundefinedorrelax` Dieses Makro arbeitet wie `\@ifundefined`, definiert dabei aber ein undefiniertes Makro nicht als `\relax`. Es muss bereits hier definiert werden, weil es gleich verwendet wird.

```
⟨*base⟩
\newcommand{\scr@ifundefinedorrelax}[1]{%
  \ifcsname #1\endcsname
    \expandafter\ifx\csname #1\endcsname\relax
      \expandafter\expandafter\expandafter\@firstoftwo
    \else
      \expandafter\expandafter\expandafter\@secondoftwo
    \fi
  \else
    \expandafter\@firstoftwo
  \fi
}
⟨/base⟩
```

2.6 Definition von KOMA-keyval-Schlüsseln (Basis)

Üblicherweise verwendet KOMA-Script für alle keyval-Schlüssel die gleiche Bereichskennung „`KOMA`“. Dadurch ist es möglich, dass Pakete die Schlüssel der Klassen umdefinieren und dass jeder Teil von KOMA-Script die Schlüssel jedes anderen Teils ausführen kann – sobald sie definiert sind.

\DefineFamily
\DefineFamilyMember
\DefineFamilyKey
\scr@if@family
\scr@if@familymember

Familie, Familienmitglied und keyval-Schlüssel für ein Familienmitglied definieren. Dabei wird als Familienmitglied per Default der (Datei-)Name des aktuellen Pakets bzw. der aktuellen Klasse verwendet.

```
⟨*base⟩
\newcommand*{\DefineFamily}[1]{%
  \scr@ifundefinedorrelax{#1@key@familylist}{%
    \@namedef{#1@key@familylist}{}%
  }{}%
}
\newcommand*{\scr@if@family}[2]{%
  \scr@ifundefinedorrelax{#1@key@familylist}{%
    \PackageError{scrbase}{%
      unknown family ‘#1’%
    }{%
      You’ve tried to #2 for family ‘#1’.\MessageBreak
      This my be done only for previous defined families.\MessageBreak
      You should call ‘\string\DefineFamily{#1}’ first.%
    }%
  }%
}
\newcommand*{\DefineFamilyMember}[2][.\@currname.\@currext]{%
  \scr@if@family{#2}{define a family member}{%
    \edef\reserved@a{%
      \noexpand\in@{#2#1,}{\csname #2@key@familylist\endcsname}}%
    \reserved@a
    \ifin@\else
      \expandafter\edef\csname #2@key@familylist\endcsname{%
        \@nameuse{#2@key@familylist}#2#1,}%
    \fi
  }%
}
\newcommand*{\scr@if@familymember}[3]{%
  \scr@if@family{#2}{#3}{%
    \edef\reserved@a{%
      \noexpand\in@{#2#1,}{\csname #2@key@familylist\endcsname}}%
    \reserved@a
    \ifin@
      \expandafter\@firstofone
    \else
      \PackageError{scrbase}{%
        unknown member ‘#1’ at family ‘#2’%
      }{%
        You’ve tried to #3 for member ‘#1’ of family ‘#2’.\MessageBreak
        This my be done only for previous defined family members.\MessageBreak
        You should call ‘\string\DefineFamilyMember[{#1}]{#2}’ first.%
      }%
      \expandafter\@gobble
    \fi
  }%
```

```
}
\newcommand*{\DefineFamilyKey}[2][.\@currname.\@currext]{%
  \scr@if@familymember{#1}{#2}{define a key}{%
    \define@key{#2#1}%
  }%
}
⟨/base⟩
```

\KOMA@key
\KOMA@key@familylist

keyval-Schlüssel für KOMA-Script werden mit \KOMAkey definiert, damit auch die Familie korrekt gesetzt und ggf. der Liste der Familien (\KOMA@key@familylist) hinzugefügt wird.

```
⟨*koma⟩
\newcommand*{\KOMA@key}[1][.\@currname.\@currext]{%
  \DefineFamilyMember{KOMA}%
  \DefineFamilyKey[#1]{KOMA}%
}
⟨/koma⟩
```

2.7 Optionen

Ab Version 2.97c sollen sowohl globale KOMA-Optionen als auch die KOMA-Optionen der jeweiligen Datei als auch die normalen Optionen in der Reihenfolge ausgeführt werden, in der sie angegeben wurden – zuerst die globalen Optionen, dann die lokalen. Dabei werden nur die Optionen verwendet, die auch für das jeweilige Paket bzw. die jeweilige Klasse definiert wurden.

\@globaloptionslist

Die Lister der globalen Optionen wird teilweise benötigt. Weil *xkeyval* dämlicherweise glaubt, es müsse alle Optionen mit Gleichheitszeichen aus der Liste der Klassenoptionen entfernen (seltsamerweise aber nicht die (x)keyval-Optionen, die ohne Wert aufgerufen werden, weil ihr Defaultwert verwendet werden soll), wird hier ein eigenes Makro definiert.

```
⟨*base⟩
\newcommand*{\@globaloptionslist}{}
\let\@globaloptionslist\@classoptionslist
\scr@ifundefinedorrelax{XKV@classoptionslist}{}{%
  \ifx\XKV@classoptionslist\@empty\else
    \let\@globaloptionslist\XKV@classoptionslist
  \fi
}
⟨/base⟩
```

\FamilyProcessOptions

Diese Anweisung führt alle Optionen, sowohl normal deklarierte als auch keyval-Optionen für eine angegebene Familie (2. Argument) und das aktuelle Mitglied (1. optionales Argument) in der Reihenfolge, in der sie verwendet wurden aus – zuerst globale Optionen, dann lokale Optionen. Das entspricht der Verwendung von \ProcessOptions* mit dem Unterschied,

dass eben auch die keyval-Optionen abgearbeitet werden. keyval-Optionen, die keinem Mitglied zugeordnet, sondern universell sind, werden dabei vor gleichnamigen keyval-Optionen des aktuellen Mitglieds ausgeführt.

```
⟨*base⟩
\newcommand*{\FamilyProcessOptions}[2][.\@currname.\@currext]{%
  \let\ds@\@empty
  \edef\@curroptions{\@ptionlist{\@currname.\@currext}}%
  \ifx\@currext\@clsextension\else
    \@for\CurrentOption:=\@globaloptionslist\do{%
      \ifx\CurrentOption\@empty\else
        \@expandtwoargs\in@{,\CurrentOption,}{,\@declaredoptions,}%
        \ifin@
          \@use@ption
          \expandafter\let\csname ds@\CurrentOption\endcsname\@empty
        \else
          \scr@optiontest{#2}\CurrentOption{}{%
            \@use@ption
            \@expandtwoargs\setkeys{#2}%
            \CurrentOption
          }%
          \scr@optiontest{#2#1}\CurrentOption{}{%
            \@use@ption
            \@expandtwoargs\setkeys{#2#1}%
            \CurrentOption
          }%
        \fi
      \fi
    }%
  \fi
  \@for\CurrentOption:=\@curroptions\do{%
    \@use@ption
    \scr@ifundefinedorrelax{ds@\CurrentOption}{%
      \scr@optiontest{#2}\CurrentOption{%
        \scr@optiontest{#2#1}\CurrentOption{%
          \default@ds
        }{%
          \edef\reserved@a{%
            \noexpand\setkeys{#2#1}{\CurrentOption}%
            \noexpand\def\noexpand\CurrentOption{\CurrentOption}%
          }\reserved@a
        }%
      }{%
        \@expandtwoargs\setkeys{#2}%
        \CurrentOption
        \scr@optiontest{#2#1}\CurrentOption{}{%
          \@expandtwoargs\setkeys{#2#1}%
          \CurrentOption
        }%
      }%
```

```
    }{%
    }%
  }%
  \@for\CurrentOption:=\@declaredoptions\do{%
    \expandafter\let\csname ds@\CurrentOption\endcsname\relax}%
  \let\CurrentOption\@empty
  \let\@fileswith@pti@ns\@@fileswith@pti@ns
  \AtEndOfPackage{\let\@unprocessedoptions\relax}%
}
\@onlypreamble\FamilyProcessOptions
```

`\scr@optiontest`
`\scr@pti@ntest`

Es werden zwei interne Makros benötigt, um zu testen, ob eine entsprechende keyval-Option definiert ist. Grund der Übung ist, dass sowohl das Familienargument als auch die Option selbst expandiert werden muss.

```
\newcommand*{\scr@optiontest}[2]{%
  \edef\reserved@a{\noexpand\scr@pti@ntest{#1}#2=\noexpand\@nil}%
  \reserved@a
}
\newcommand*{\scr@pti@ntest}{}
\def\scr@pti@ntest#1#2=#3\@nil{\scr@ifundefinedorrelax{KV@#1@#2}}
⟨/base⟩
```

`\KOMAProcessOptions`

Diese Anweisung führt alle Optionen, sowohl normal deklarierte als auch keyval-Optionen eines Pakets bzw. einer Klasse in der Reihenfolge, in der sie verwendet wurden aus – zuerst globale Optionen, dann lokale Optionen. Das entspricht der Verwendung von `\ProcessOptions*` mit dem Unterschied, dass eben auch die keyval-Optionen abgearbeitet werden. keyval-Optionen, die keiner einzelnen Klasse und keinem einzelnen Paket zugeordnet werden, werden dabei vor gleichnamigen eigenen keyval-Optionen ausgeführt.

```
⟨*koma⟩
\newcommand*{\KOMAProcessOptions}{\FamilyProcessOptions{KOMA}}
\@onlypreamble\KOMAProcessOptions
⟨/koma⟩
```

`internalonly`
`\dont@let@as@internal@defined`

Anweisungen, die mit einem internen und einem Anwendername vorhanden sind, können vor dem Anwender optional verborgen werden. Wird kein Argument angegeben, so werden alle Anweisungen verborgen.

```
⟨*base⟩
\newcommand*{\dont@let@as@internal@defined}{/}
\newif\if@let@as@internalonly@defined\@let@as@internalonly@definedfalse
\DefineFamily{KOMA}
\DefineFamilyMember{KOMA}
\DefineFamilyKey{KOMA}{internalonly}[\relax]{%
  \ifx\relax#1\relax
    \@let@as@internalonly@definedtrue
  \else
```

```
    \@let@as@internalonly@definedfalse
    \begingroup
      \@expandtwoargs\in@{#1}\dont@let@as@internal@defined
    \expandafter\endgroup
    \ifin@\else\g@addto@macro\dont@let@as@internal@defined{#1/}\fi
  \fi
}
⟨/base⟩
```

```
⟨koma⟩\KOMAProcessOptions\relax
⟨base⟩\FamilyProcessOptions{KOMA}\relax
```

2.8 Ein wenig zusätzliche Mathematik für ε-TEX

`\XdivY` Ergebnis ist erster Argument durch zweites abgerundet auf die nächst kleinere ganze Zahl.

```
⟨*base⟩
\newcommand*{\XdivY}[2]{%
  \numexpr ( #1 + #2 / 2 ) / #2 - 1\relax
}
⟨/base⟩
```

`\XmodY` Ergebnis ist erstes Argument modulo zweites Argument.

```
⟨*base⟩
\newcommand*{\XmodY}[2]{%
  \numexpr #1 - #2 * \XdivY{#1}{#2}\relax
}
⟨/base⟩
```

2.9 Häufig benötigte Befehle

`\if@atdocument` `\@atdocumenttrue` `\@atdocumentfalse` Dieser Schalter wird bei `\begin{document}` wahr. Er wird von KOMA-Script für Befehle verwendet, die sich unterschiedlich verhalten, je nachdem, ob sie in der Präambel oder danach aufgerufen werden.

```
⟨*base⟩
\newif\if@atdocument
\AtBeginDocument{\@atdocumenttrue}
\@onlypreamble\@atdocumentrue
\@onlypreamble\@atdocumentfalse
⟨/base⟩
```

`\let@as@internal@defined` Dieses Makro wird intern verwendet, um dokumentierte Makros mit der Definition von internen Makros zu versehen, falls sie nicht bereits definiert oder als nicht zu definieren markiert sind. Der Name des dokumentierten Makros wird als Parameter übergeben. Dies wird gleich benötigt.

```
⟨*base⟩
\newcommand*{\let@as@internal@defined}[1]{%
```

```
\begingroup
  \if@let@as@internalonly@defined
    \in@true
  \else
    \@expandtwoargs\in@{/#1/}\dont@let@as@internal@defined
  \fi
\expandafter\endgroup
\ifin@
  \PackageInfo{%
    scrbase%
  }{%
    \expandafter\string\csname #1\endcsname\space excluded from being
    defined%
  }%
\else
  \@ifundefined{#1}{%
    \begingroup
      \expandafter\def\expandafter\@tempa
        \expandafter{\csname scr@#1\endcsname}
      \expandafter\def\expandafter\@tempb
        \expandafter{\csname #1\endcsname}
    \expandafter\expandafter\expandafter\endgroup
    \expandafter\expandafter\expandafter\let\expandafter\@tempb\@tempa
    \if@atdocument\else
      \g@addto@macro\document{%
        \expandafter\ifx\csname #1\expandafter\endcsname
        \csname scr@#1\endcsname
        \else
          \PackageWarning{scrbase}{%
            \expandafter\string\csname #1\endcsname\space was
            redefined\MessageBreak
            at the document preamble.\MessageBreak
            If you did not redefine it by yourself, have a\MessageBreak
            a look at the description of this command at\MessageBreak
            the manual of every package you are using\@gobble
          }%
        \fi
      }%
    \fi
  }{%
    \PackageInfo{%
      scrbase%
    }{%
      \expandafter\string\csname #1\endcsname\space already
      defined.\MessageBreak
      If \expandafter\string\csname #1\endcsname\space does not behave
      like\MessageBreak
      it was described at the KOMA-Script manual, try to\MessageBreak
      load ‘scrbase.sty’ earlier or find out at which\MessageBreak
```

```
        package or class \expandafter\string\csname #1\endcsname\space
        was defined before\@gobble
      }%
    }%
  \fi
}
⟨/base⟩
```

\ifundefinedorrelax Dieses Makro arbeitet wie `\@ifundefined`, definiert dabei aber ein undefiniertes Makro nicht als `\relax`.

```
⟨*base⟩
\let@as@internal@defined{ifundefinedorrelax}
⟨/base⟩
```

\ifnotundefined Dieses Makro arbeitet genau umgekehrt zu `\ifundefined`. Manchmal ist es ganz nützlich, auch das zu haben. Dieses Makro heißt nicht `\ifdefined`, weil ε-TEX bereits ein solches Primitiv bietet.

```
⟨*base⟩
\newcommand{\ifnotundefined}[1]{%
  \ifcsname #1\endcsname
    \expandafter\@firstoftwo
  \else
    \expandafter\@secondoftwo
  \fi
}
⟨/base⟩
```

\ifstr Dieses Makro vergleicht zwei Tokenfolgen. Eigentlich müsste es also eher `\ifequal` oder ähnlich heißen.

```
⟨*base⟩
\newcommand\ifstr[2]{%
  \begingroup\protected@edef\@tempa{#1}\protected@edef\@tempb{#2}%
  \ifx\@tempa\@tempb
    \endgroup\expandafter\@firstoftwo
  \else
    \endgroup\expandafter\@secondoftwo
  \fi
}
⟨/base⟩
```

\ifnumber Dieses Makro testet, ob ein übergebener Parameter eine positive, ganze Zahl ist. Ist dies der Fall, wird der zweite Parameter ausgeführt, sonst der dritte.

```
⟨*base⟩
\newcommand\ifnumber[1]{%
  \begingroup\@tempswafalse\let\scr@next\test@number
  \expandafter\scr@next#1\scr@next
  \if@tempswa
```

```
    \endgroup\expandafter\@firstoftwo
  \else
    \endgroup\expandafter\@secondoftwo
  \fi
}
```

\test@number Für den eigentlichen Test wird der erste Parameter Token für Token gescannt und beurteilt.

```
\newcommand*{\test@number}[1]{%
  \ifx \scr@next#1
    \let\scr@next\relax
  \else
```

Dabei wird für einen schnelleren Vergleich die Tatsache genutzt, dass Ziffernzeichen immer im selben durchgängigen Bereich liegen

```
    \@tempcnta=\expandafter\expandafter\expandafter\number
    \expandafter`#1\relax
    \ifnum \@tempcnta>47
      \ifnum \@tempcnta<58
        \@tempswatrue
      \else\@tempswafalse\fi
    \else\@tempswafalse\fi
```

Das zeichenweise Scannen und Parsen kann natürlich nicht einfach abgebrochen werden. Stattdessen ist der Parameterrest noch zu entfernen.

```
    \if@tempswa\else\let\scr@next\gobble@till@scr@next\fi
  \fi
  \scr@next
}
```

\gobble@tillscr@@next Das geschieht mit einem einzigen „Leermacro“.

```
\newcommand*{\gobble@till@scr@next}{}
\def\gobble@till@scr@next#1\scr@next{}
⟨/base⟩
```

\ifdimen Dieses Makro testet, ob ein übergebener Parameter eine Dimension ist und führt in Abhängigkeit davon den zweiten oder dritten Parameter aus.

```
⟨*base⟩
\newcommand{\ifdimen}[1]{%
  \begingroup\@tempswatrue\let\scr@next\test@posdimen
  \expandafter\test@dimen#1\scr@next
  \if@tempswa
    \endgroup\expandafter\@firstoftwo
  \else
    \endgroup\expandafter\@secondoftwo
  \fi
}
```

\test@dimen Als erste muss ein Vorzeichen entfernt werden.

```
\newcommand*{\test@dimen}[1]{%
  \ifx -#1\else\ifx +#1\else
      \def\scr@next{\let\scr@next\test@posdimen\scr@next#1}\fi\fi
  \scr@next
}
```

\test@posdimen

Dann muss getestet werden, ob es sich beim Rest um eine Dimension handelt. Dabei werden der Einfachheit halber mehrere Dezimalkomma und -punkte erlaubt. Natürlich könnte man es auch wirklich richtig machen, aber das wäre nochmal etwas aufwendiger. Ist das wirklich notwendig?

```
\newcommand*{\test@posdimen}[1]{%
  \ifx \scr@next#1%
    \@tempswafalse\let\scr@next\relax
  \else
    \if .#1\else\if ,#1\else
        \@tempcnta=%
        \expandafter\expandafter\expandafter\number\expandafter`#1\relax
        \ifnum \@tempcnta>47
          \ifnum \@tempcnta<58
          \else\def\scr@next{\test@dimunt#1}\fi
        \else\def\scr@next{\test@dimunt#1}\fi
    \fi\fi
  \fi
  \scr@next
}
```

\testdimunt
\test@trueunt

Außer einer Gleitkomma- bzw. Gleitpunktzahl muss auch noch die Einheit behandelt werden. Dabei sind sowohl die neun Standardeinheiten als auch deren „true“-Variante zu berücksichtigen. Deshalb wird zuerst ein eventuelles „`true`“ weggeworfen.

```
\newcommand*{\test@dimunt}{}
\def\test@dimunt#1\scr@next{%
  \expandafter\test@trueunt#1truetrue\test@trueunt
}
\newcommand*{\test@trueunt}{}
\def\test@trueunt#1true#2true#3\test@trueunt{%
  \edef\@tempa{#1}
  \ifx\@tempa\@empty\test@@ifdimunt{#2}\else\test@@ifdimunt{#1}\fi
}
```

\test@@ifdimunt

Anschließend wird dann getestet, ob es sich um eine der neun üblichen Einheiten handelt.

```
\newcommand*\test@@ifdimunt[1]{%
  \@tempswafalse
  \ifstr{#1}{pt}{\@tempswatrue}{%
    \ifstr{#1}{pc}{\@tempswatrue}{%
      \ifstr{#1}{in}{\@tempswatrue}{%
        \ifstr{#1}{bp}{\@tempswatrue}{%
```

```
          \ifstr{#1}{cm}{\@tempswatrue}{%
            \ifstr{#1}{mm}{\@tempswatrue}{%
              \ifstr{#1}{dd}{\@tempswatrue}{%
                \ifstr{#1}{cc}{\@tempswatrue}{%
                  \ifstr{#1}{sp}{\@tempswatrue}{}%
                }%
              }%
            }%
          }%
        }%
      }%
    }%
  }%
}
⟨/base⟩
```

`\ifpdftex` `\scr@ifpdftex` If-then-else-Konstrukt, um zu testen, ob pdfTEX verwendet wird.

```
⟨*base⟩
\newcommand{\scr@ifpdftex}{\ifnotundefined{pdftexversion}}
\let@as@internal@defined{ifpdftex}
\BeforePackage{pdfcprot}{\let\ifpdftex\relax}
\AfterPackage{pdfcprot}{\let@as@internal@defined{ifpdftex}}
⟨/base⟩
```

`\ifVTeX` `\scr@ifVTeX` If-then-else-Konstrukt, um zu testen, ob VTEX verwendet wird.

```
⟨*base⟩
\newcommand{\scr@ifVTeX}{\ifnotundefined{VTeXversion}}
\let@as@internal@defined{ifVTeX}
⟨/base⟩
```

`\ifpdfoutput` `\scr@ifpdfoutput` If-then-else-Konstrukt, um abhängig davon, ob als `pdf` ausgegeben wird oder nicht, Befehle auszuführen. Dieses Makro wird für die Deklaration der Option `pagesize` benötigt.

```
⟨*base⟩
\newcommand{\scr@ifpdfoutput}{%
  \scr@ifundefinedorrelax{pdfoutput}{%
    \scr@ifundefinedorrelax{OpMode}{%
      \expandafter\@secondoftwo%
    }{%
      \ifnum\OpMode=1
        \expandafter\@firstoftwo
      \else
        \expandafter\@secondoftwo
      \fi
    }%
  }{%
    \ifnum\pdfoutput>0
      \expandafter\@firstoftwo
    \else
```

```
      \expandafter\@secondoftwo
    \fi
  }%
}%
\let@as@internal@defined{ifpdfoutput}
⟨/base⟩
```

`\ifpsoutput` `\scr@ifpsoutput` If-then-else-Konstrukt, um abhängig davon, ob früher oder später `ps` ausgegeben wird oder nicht, Befehle auszuführen.

```
⟨*base⟩
\newcommand*{\scr@ifpsoutput}{%
  \ifnotundefined{if@dvips}{%
    \if@dvips
      \expandafter\@firstoftwo
    \else
      \expandafter\@secondoftwo
    \fi
  }{%
    \ifnotundefined{OpMode}{%
      \ifnum\OpMode=2
        \expandafter\@firstoftwo
      \else
        \expandafter\@secondoftwo
      \fi
    }{%
      \expandafter\@secondoftwo
    }%
  }%
}
\let@as@internal@defined{ifpsoutput}
⟨/base⟩
```

`\ifdvioutput` `\scr@ifdvioutput` If-then-else-Konstrukt, um abhängig davon, ob `dvi` ausgegeben wird oder nicht, Befehle auszuführen.

```
⟨*base⟩
\newcommand*{\scr@ifdvioutput}{%
  \scr@ifundefinedorrelax{pdfoutput}{%
    \scr@ifundefinedorrelax{OpMode}{%
      \expandafter\@firstoftwo
    }{%
      \ifnum\OpMode=0
        \expandafter\@firstoftwo
      \else
        \expandafter\@secondoftwo
      \fi
    }%
  }{%
    \ifnum\pdfoutput=0
      \expandafter\@firstoftwo
```

```
    \else
      \expandafter\@secondoftwo
    \fi
  }%
}
\let@as@internal@defined{ifdvioutput}
⟨/base⟩
```

\scr@smashdp Gibt Material aus, erzeugt dabei aber nur Höhe und Breite, jedoch keine Tiefe.

```
⟨*base⟩
\newcommand*{\scr@smashdp}[1]{%
  \begingroup
    \def\finsm@sh{\dp\z@\z@\box\z@}%
    \ifmmode
      \expandafter\mathpalette\expandafter\mathsm@sh
    \else
      \expandafter\makesm@sh
    \fi
    {#1}%
  \endgroup
}
⟨/base⟩
```

\PackageInfoNoLine Arbeitet wie \PackageInfo aber ohne Ausgabe der Zeilennummer.

```
⟨*base⟩
\providecommand*{\PackageInfoNoLine}[2]{%
  \PackageInfo{#1}{#2\@gobble}%
}
⟨/base⟩
```

\ClassInfoNoLine Arbeitet wie \ClassInfo aber ohne Ausgabe der Zeilennummer.

```
⟨*base⟩
\providecommand*{\ClassInfoNoLine}[2]{%
  \ClassInfo{#1}{#2\@gobble}%
}
⟨/base⟩
```

2.10 Definition von KOMA-keyval-Schlüsseln (Hauptteil)

\FamilyExecuteOptions Entspriche \ExecuteOptions jedoch keyval-Optionen.

```
⟨*base⟩
\newcommand*{\FamilyExecuteOptions}[3][.\@currname.\@currext]{%
  \def\reserved@a##1\@nil{%
    \@for\scr@key@atlist:=#3\do{%
      \expandafter\scr@key@split@name@value\scr@key@atlist==\@nil
      \scr@ifundefinedorrelax{KV@#2@\scr@key@name}{}{%
        \protected@edef\reserved@a{\noexpand\setkeys{#2}{\scr@key@atlist}%
```

```
          \noexpand\def\noexpand\scr@key@atlist{\scr@key@atlist}%
          \noexpand\def\noexpand\scr@key@name{\scr@key@name}%
        }%
        \reserved@a
      }%
      \scr@ifundefinedorrelax{KV@#2#1@\scr@key@name}{}{%
        \protected@edef\reserved@a{%
          \noexpand\setkeys{#2#1}{\scr@key@atlist}%
          \noexpand\def\noexpand\scr@key@atlist{\scr@key@atlist}%
          \noexpand\def\noexpand\scr@key@name{\scr@key@name}%
        }%
        \reserved@a
      }%
      \let\CurrentOption\scr@key@atlist
      \csname ds@\CurrentOption\endcsname
    }%
    \edef\CurrentOption{##1}%
  }%
  \expandafter\reserved@a\CurrentOption\@nil
}
\@onlypreamble\FamilyExecuteOptions
⟨/base⟩
```

`\KOMAExecuteOptions` Liste von Optionen ausführen

```
⟨*koma⟩
\newcommand*{\KOMAExecuteOptions}{\FamilyExecuteOptions{KOMA}}
\@onlypreamble\KOMAExecuteOptions
⟨/koma⟩
```

`\FamilyOptions` Es werden Schlüssel einer Familie verwendet. Existiert ein Schlüssel nicht, so wird eine Fehlermeldung ausgegeben.

```
⟨*base⟩
\newcommand*{\FamilyOptions}[2]{%
  \Family@Options{#1}{#2}{%
    \PackageError{scrbase}{%
      option ‘\scr@key@name’ unknown at family ‘#1’%
    }{%
      You have used \string\FamilyOptions{#1} to set
      ‘\scr@key@atlist’,\MessageBreak
      but family ‘#1’ does not know any option named
      ‘\scr@key@name’.\MessageBreak
      See the manual for the package or class using family ‘#1’ for
      more\MessageBreak
      informations about options and their values.%
    }%
  }%
}
% \begin{macro}{\Family@Options}
%   \changes{v3.00}{2008/04/30}{Neues Makro (intern)}
```

```
%   \changes{v3.04b}{2010/01/29}{\cs{edef} durch \cs{protected@edef} ersetzt}
% Das interne Macro \cs{Family@Options} macht die eigentliche Arbeit. Das
% erste Argument ist dabei die Familie. Das zweite Argument ist die Liste der
% Optionen, das zweite Code, der im Fehlerfall ausgeführt werden soll. Dabei
% sind die Makros \cs{scr@key@atlist}, \cs{scr@key@name} und
% \cs{scr@key@value} gültig und können verwendet werden.
%
% Anmerkung: Es wird nicht auf \textsf{xkeyval} und den dortigen Befehl
% \cs{setkeys+} gesetzt, weil dieser die Optionen nicht in der Reihenfolge der
% Optionen, sondern in der Reihenfolge der Familien ausführt. Das passt mir
% aber hier gar nicht.
%
% In einer Schleife über alle Optionen werden die Optionen zunächst in Option
% und Wert aufgespaltet.
%    \begin{macrocode}
\newcommand*{\Family@Options}[3]{%
  \@for\scr@key@atlist:=#2\do{%
```

Auf vielfachen Wunsch und entgegen meiner Überzeugung, sind auch leere Argumente in der Optionenliste erlaubt. Die erste Form wäre, wenn zwei Kommas diret aufeinander folgen:

```
    \ifx\scr@key@atlist\@empty\else
```

Die zweite Möglichkeit wäre, dass noch Leerzeichen zwischen den Komma übrig bleiben:

```
      \ifx\scr@key@atlist\space\else
        \expandafter\scr@key@split@name@value\scr@key@atlist==\@nil
```

Die dritte Möglichkeit, dass auch noch ein Gleichheitszeichen übrig gelassen wurde, werde ich aber nicht abfangen. Irgendwann müssen die Anwender auch mal ihre Fehler einsehen und korrigieren!

Dann wird in einer Schleife über alle Familien festgestellt, ob die Option in einer Familie bekannt ist und ggf. ausgeführt.

```
        \scr@key@found@false
        \edef\@tempa{\csname #1@key@familylist\endcsname}%
        \@for\@tempa:=\@tempa\do{%
          \scr@ifundefinedorrelax{KV@\@tempa @\scr@key@name}{}{%
            \scr@key@found@true
            \protected@edef\reserved@a{%
              \noexpand\setkeys{\@tempa}{\scr@key@atlist}%
              \noexpand\def\noexpand\@tempa{\@tempa}%
              \noexpand\def\noexpand\scr@key@atlist{\scr@key@atlist}%
              \noexpand\def\noexpand\scr@key@name{\scr@key@name}%
              \noexpand\def\noexpand\scr@key@value{\scr@key@value}%
            }%
            \reserved@a
          }%
        }%
        \ifscr@key@found@\else#3\fi
      \fi
```

```
    \fi
  }%
}
```

\ifscr@key@found@
\scr@key@atlist
\scr@key@name
\scr@key@value
\scr@key@split@name@value

```
\newif\ifscr@key@found@
\newcommand*{\scr@key@atlist}{}
\newcommand*{\scr@key@name}{}
\newcommand*{\scr@key@value}{}
\newcommand*{\scr@key@split@name@value}{}
\def\scr@key@split@name@value#1=#2=#3\@nil{%
  \KV@@sp@def\scr@key@name{#1}%
  \KV@@sp@def\scr@key@value{#2}%
}
⟨/base⟩
```

\FamilyOption
\Family@Option

Im Gegensatz zu `\FamilyOptions` kann man mit dieser Anweisung nur eine einzige Option setzen. Dafür kann man gleich mehrere durch Komma getrennte Argumente für diese Option setzen. Das erste obligatorische Argument ist wieder die Familie, das zweite obligatorische Argument ist die Option, das dritte, obligatorische Argument ist die Liste der Werte für diese Option.

```
⟨*base⟩
\newcommand*{\FamilyOption}[3]{%
  \Family@Option{#1}{#2}{#3}{%
    \PackageError{scrbase}{%
      unknown option ‘\scr@key@name’ for familie ‘#1’%
    }{%
      You have used \string\FamilyOption{#1} to set
      ‘\scr@key@name’,\MessageBreak
      but family ’#1’ does not know any option named
      ‘\scr@key@name’.\MessageBreak
      See the the manual of the class or package, that provides
      family\MessageBreak
      ‘#1’ for more informations about options and their values.%
    }%
  }%
}
\newcommand*{\Family@Option}[4]{%
  \expandafter\KV@@sp@def\expandafter\scr@key@name\expandafter{#2}%
  \scr@key@found@false
  \edef\@tempa{\csname #1@key@familylist\endcsname}%
  \@for\@tempa:=\@tempa\do{%
    \scr@ifundefinedorrelax{KV@\@tempa @\scr@key@name}{}{%
      \scr@key@found@true
      \let\@tempb\@tempa
    }%
  }%
  \ifscr@key@found@
```

```
    \let\reserved@a\@empty
    \@for\@tempa:=#3\do{%
      \expandafter\KV@@sp@def\expandafter\@tempa\expandafter{\@tempa}%
```

Auf vielfachen Wunsch und entgegen meiner Überzeugung, sind auch leere Argumente in der Werteliste erlaubt:

```
      \ifx\@tempa\@empty\else
        \protected@edef\reserved@a{\reserved@a,\scr@key@name=\@tempa}%
      \fi
    }%
    \ifx\reserved@a\@empty
      \protected@edef\reserved@a{\noexpand\setkeys{\@tempb}{\scr@key@name}}%
    \else
      \protected@edef\reserved@a{%
        \noexpand\setkeys{\@tempb}{\expandafter\@gobble\reserved@a}}%
    \fi
    \reserved@a
  \else
    #4
  \fi
}
⟨/base⟩
```

`\KOMAoptions` Es werden Schlüssel verwendet und ggf. danach noch Dinge ausgeführt.

```
⟨*koma⟩
\newcommand*{\KOMAoptions}[1]{%
  \let\post@kopt@hook\@empty
  \KOMA@options{#1}{%
```

Wenn der Schlüssel für keines der bekannten Familienmitglieder definiert ist, wird ein Fehler ausgegeben.

```
    \PackageError{scrkbase}{%
      unknown KOMA option ‘\scr@key@name’%
    }{%
      You have used \string\KOMAoptions\space to set
      ‘\scr@key@atlist’,\MessageBreak
      but KOMA-Script does not know any option named
      ‘\scr@key@name’.\MessageBreak
      See the KOMA-Script manual for more informations about options
      and\MessageBreak
      their values.%
    }%
  }%
  \post@kopt@hook\let\post@kopt@hook\relax
}
```

`\KOMA@options` Das interne Macro `\KOMA@options` macht die eigentliche Arbeit. Das erste Argument ist dabei die Liste der Optionen, das zweite Code, der im Fehlerfall ausgeführt werden soll. Dabei sind die Makros `\KOMA@key@atlist`,

\KOMA@key@name und \KOMA@key@value gültig und können verwendet werden.

Anmerkung: Es wird nicht auf ykeyval und den dortigen Befehl \setkeys+ gesetzt, weil dieser die Optionen nicht in der Reihenfolge der Optionen, sondern in der Reihenfolge der Familien ausführt. Das passt mir aber hier gar nicht.

In einer Schleife über alle Optionen werden die Optionen zunächst in Option und Wert aufgespaltet.

```
\newcommand*{\KOMA@options}{\Family@Options{KOMA}}
⟨/koma⟩
```

\post@kopt@hook
\AfterKOMAoptions

Eine Option kann dafür sorgen, dass nach den Optionen noch etwas passiert.

```
⟨*koma⟩
\newcommand*{\post@kopt@hook}{}
\let\post@kopt@hook\relax
\newcommand*{\AfterKOMAoptions}{%
  \ifx\post@kopt@hook\relax
    \PackageWarning{scrkbase}{%
      \string\AfterKOMAoptions\space can only be used inside a\MessageBreak
      KOMA-Script option that is called using \string\KOMAoptions.\MessageBreak
      Because of this, command will be ignored%
    }%
    \expandafter\@gobbletwo
  \else
    \expandafter\l@addto@macro
  \fi
  \post@kopt@hook
}
⟨/koma⟩
```

\l@addto@macro

Damit \AfterKOMAoptions lokal möglich ist, wird eine lokal arbeitende Variante von \g@addto@macro benötigt, wie sie hier vorgestellt wird.

```
⟨*base⟩
\scr@ifundefinedorrelax{unexpanded}{%
  \newcommand*{\g@addto@macro@temp}{}%
  \newcommand{\l@addto@macro}[2]{%
    \begingroup%
      \toks@\expandafter{#1#2}%
      \xdef\g@addto@macro@temp{\the\toks@}%
    \endgroup
    \let#1\g@addto@macro@temp
  }%
}{%
  \newcommand{\l@addto@macro}[2]{%
    \edef#1{\unexpanded\expandafter{#1#2}}%
  }%
}
```

```
⟨/base⟩
```

\KOMAoption Im Gegensatz zu \KOMAoptions kann man mit dieser Anweisung nur eine einzige Option setzen. Dafür kann man gleich mehrere durch Komma getrennte Argumente für diese Option setzen. Das erste obligatorische Argument ist die Option, das zweite, obligatorische Argument ist die Liste der Werte für diese Option.

```
⟨*koma⟩
\newcommand*{\KOMAoption}[2]{%
  \let\post@kopt@hook\@empty
  \Family@Option{KOMA}{#1}{#2}{%
    \PackageError{scrkbase}{%
      unknown KOMA option ‘\scr@key@name’%
    }{%
      You have used \string\KOMAoption\space to set
      ‘\scr@key@name’,\MessageBreak
      but KOMA-Script does not know any option named
      ‘\scr@key@name’.\MessageBreak
      See the KOMA-Script manual for more informations about options
      and\MessageBreak
      their values.%
    }%
  }%
  \post@kopt@hook\let\post@kopt@hook\relax
}
⟨/koma⟩
```

\FamilyUnknownKeyValue Dieses Makro berichtet über Fehler wegen unbekannter Wertübergabe an einen Schlüssel. Das erste Argument ist die Familie, das zweit der Schlüssel, das dritte der Wert und das vierte die Liste der möglichen Werte.

```
⟨*base⟩
\newcommand*{\FamilyUnknownKeyValue}[4]{%
  \PackageError{%
    scrbase%
  }{%
    option ‘#2’ of family ‘#1’ has no value ‘#3’%
  }{%
    You’ve tried to set option ‘#2’ of family ’#1’ to value ‘#3’.\MessageBreak
    But allowed values are only:\MessageBreak
    #4\FamilyElseValues.\MessageBreak
    See the manual of the class or package, that provides family
    ‘#1’\MessageBreak
    for more information about options and their values.%
  }%
}
⟨/base⟩
```

\FamilyElseValues Diese Anweisung kann vor dem Setzen eines Wertes definiert werden, um bei einer etwaigen Fehlermeldung weitere Werte als erlaubt auszugeben. Am

Ende von \FamilySetBool und \FamilySetNumerical wird die Anweisung automatisch zu \@empty. Es ist zu beachten, dass bei der Definition der Anweisung als erstes ein Komma und ein Leerzeichen zu setzen ist!

```
⟨*base⟩
\newcommand*{\FamilyElseValues}{}
⟨/base⟩
```

\FamilyBoolKey Dieses Makro definiert einen Schlüssel für eine Familie. Dieser Schlüssel ist ein Schalter, der über diverse Werte geschaltet werden kann. Das erste, optionale Argument ist das Familienmitglied, das zweite, obligatorische Argument ist die Familie, das dritte der Name des Schlüssels, das vierte der Schalter.

```
⟨*base⟩
\newcommand*{\FamilyBoolKey}[4][.\@currname.\@currext]{%
  \@ifundefined{if#4}{%
    \expandafter\newif\csname if#4\endcsname
  }{}%
  \DefineFamilyKey[{#1}]{#2}{#3}[true]{\FamilySetBool{#2}{#3}{#4}{##1}}%
}
```

\FamilySetBool Über diese Anweisung wird der Wert dann gesetzt. Dabei ist das erste Argument die Familie, das zweite der Name des Schlüsselt, das dritte der Schalter und das vierte schließlich der gewünschte Wert.

```
\newcommand*{\FamilySetBool}[4]{%
  \ifstr{#4}{true}{\@nameuse{#3true}}{%
    \ifstr{#4}{on}{\@nameuse{#3true}}{%
      \ifstr{#4}{yes}{\@nameuse{#3true}}{%
        \ifstr{#4}{false}{\@nameuse{#3false}}{%
          \ifstr{#4}{off}{\@nameuse{#3false}}{%
            \ifstr{#4}{no}{\@nameuse{#3false}}{%
              \FamilyUnknownKeyValue{#1}{#2}{#3}{%
                'true', 'on', 'yes', 'false', 'off', 'no'%
              }%
            }%
          }%
        }%
      }%
    }%
  }%
  \let\FamilyElseValues\@empty
}
⟨/base⟩
```

\KOMA@unknown@keyval Dieses Makro berichtet über Fehler wegen unbekannter Wertübergabe an einen Schlüssel. Das erste Argument ist der Schlüssel, das zweite der Wert und das dritte die Liste der möglichen Werte.

```
⟨*koma⟩
```

```
\newcommand*{\KOMA@unknown@keyval}{\FamilyUnknownKeyValue{KOMA}}
⟨/koma⟩
```

\KOMA@ifkey Dieses Makro definiert einen Schlüssel, der einen Schalter an Hand des Wertes setzt. Als Werte sind „`true`“, „`false`“, „`on`“, „`off`“, „`yes`“ und „`no`“ erlaubt. Der Default-Wert für den Schalter ist grundsätzlich „false“, so dass der Default-Wert für die Verwendung des Schküssels grundsätzlich „true“ ist. Das erste Argument ist der Name des Schlüssels, der zweite ist der Name des Schalters.

```
⟨*koma⟩
\newcommand*{\KOMA@ifkey}{%
  \DefineFamilyMember{KOMA}%
  \FamilyBoolKey{KOMA}%
}
⟨/koma⟩
```

\KOMA@set@ifkey Die eigentliche Arbeit bei der ganzen Sache wird von diesem internen Makro erledigt. Dabei wird der Name des Schlüssels zu Argument 1, der Name des Schalters zu Argument 2 und der Wert des Schlüssels wird Argument 3.

```
⟨*koma⟩
\newcommand*{\KOMA@set@ifkey}{\FamilySetBool{KOMA}}
⟨/koma⟩
```

\FamilyNumericalKey
\Family@Numerical@Key Ein ähnliches Makro definiert einen Schlüssel, der ein nummerisches Makro auf einen bestimmten Wert setzt. Dabei können auch verschiedene symbolische Werte auf denselben nummerischen Wert abgebildet werden. Das erste, optionale Argument ist das Familienmitglied, das zweite, obligatorische Argument ist die Familie, das dritte, obligatorische Argument ist der Name der Option, das vierte, optionale Artument ist der Defaultwert für den Schlüssel, das fünfte, obligatorische Argument ist der Name des Makros, das (in der Voreinstellung als 0) definiert werden soll, das sechste, obligatorische Argument ist die Liste der symbolischen und nummerischen Werte. Die Liste hat die Form: {*Symbol*}{*Wert*},{*Symbol*}{*Wert*}... Das interne Makro wird benötigt, um das vierte, optionale Argument zu ermöglichen. Dabei ist dann das erste Argument nicht mehr optional, sondern obligatorisch.

```
⟨*base⟩
\newcommand*{\FamilyNumericalKey}[3][.\@currname.\@currext]{%
  \@ifnextchar [%]
    {\Family@Numerical@Key{#1}{#2}{#3}}{\Family@Numerical@Key{#1}{#2}{#3}[]}%
}
\newcommand*{\Family@Numerical@Key}{}
\def\Family@Numerical@Key#1#2#3[#4]#5#6{%
  \@ifundefined{#5}{\@namedef{#5}{0}}%
  \ifx\relax#4\relax
    \DefineFamilyKey[{#1}]{#2}{#3}{\FamilySetNumerical{#2}{#3}{#5}{#6}{##1}}%
```

```
  \else
    \DefineFamilyKey[{#1}]{#2}{#3}[{#4}]{%
      \FamilySetNumerical{#2}{#3}{#5}{#6}{##1}}%
  \fi
}
```

\FamilySetNumerical Die eigentliche Arbeit dabei wird von diesem Makro erledigt. Dabei wird die Familie zu Argument 1, der Name des Schlüssels zu Argument 2, der Name des Makros, das definiert werden soll zu Argument 3, die Liste zu Argument 4 und der Wert, auf den gesetzt werden soll, zu Argument 5.

```
\newcommand*{\FamilySetNumerical}[5]{%
  \begingroup
    \edef\@searched{#5}%
    \let\@tempa=\@empty
    \let\@tempb=\@empty
    \@tempswafalse
    \@for\@valuelist:=#4\do{%
      \if@tempswa\else
        \edef\@tempc{\expandafter\@firstoftwo\@valuelist}%
        \edef\@tempa{\@tempa\@tempb`\@tempc'}%
        \edef\@tempb{, }%
        \ifx\@tempc\@searched
          \@tempswatrue
          \edef\@tempa{\expandafter\@secondoftwo\@valuelist}%
        \fi
      \fi
    }%
    \if@tempswa
      \edef\@tempa{\endgroup
        \noexpand\@namedef{#3}{\@tempa}%
      }%
    \else
      \FamilyUnknownKeyValue{#1}{#2}{#5}{\@tempa}%
      \let\@tempa=\endgroup
    \fi
  \@tempa
  \let\FamilyElseValues\@empty
}
⟨/base⟩
```

\KOMA@ncmdkey Ein ähnliches Makro definiert einen Schlüssel, der ein nummerisches Makro auf einen bestimmten Wert setzt. Dabei können auch verschiedene symbolische Werte auf denselben nummerischen Wert abgebildet werden. Das erste Argument ist obligatorisch und der Name des Schlüssels. Das zweite Argument ist optional und der Defaultwert für den Schlüssel. Das dritte Argument ist obligatorisch und der Name des Makros, das definiert werden soll. Der Defaultwert dafür ist 0. Das vierte Argument ist obligatorisch und die Liste der symbolischen und nummerischen Wert. Die Liste hat die

Form: {*Symbol*}{*Wert*},{*Symbol*}{*Wert*}...

```
⟨*koma⟩
\newcommand*{\KOMA@ncmdkey}{%
  \DefineFamilyMember{KOMA}%
  \FamilyNumericalKey{KOMA}%
}
⟨/koma⟩
```

\KOMA@set@ncmdkey

Die eigentliche Arbeit dabei wird wahlweise von diesem Makro erledigt. Dabei wird der Name des Schlüssels zu Argument 1, der Name des Makros, das definiert werden soll zu Argument 2, die Liste zu Argument 3 und der Wert, auf den gesetzt werden soll, zu Argument 4.

```
⟨*koma⟩
\newcommand*{\KOMA@set@ncmdkey}{\FamilySetNumerical{KOMA}}
⟨/koma⟩
```

\FamilyStringKey
\Family@String@Key
\Family@@String@@Key

Ein ähnliches Makro definiert einen Schlüssel, der ein Makro auf einen beliegigen Wert setzt. Das erste, optionale Argument ist das Familienmitglied, das zweite, obligatorische Argument ist die Familie, das dritte, obligatorische Argument ist der Name der Option, das vierte, optionale Artument ist der Defaultwert für den Schlüssel, das fünfte, obligatorische Argument ist der Name des Makros, das (in der Voreinstellung als leer) definiert werden soll. Die internen Makro werden benötigt, um das vierte, optionale Argument zu ermöglichen. Dabei ist dann das erste Argument nicht mehr optional, sondern obligatorisch.

```
⟨*base⟩
\newcommand*{\FamilyStringKey}[3][.\@currname.\@currext]{%
  \@ifnextchar [%]
    {\Family@String@Key{#1}{#2}{#3}}{\Family@@String@@Key{#1}{#2}{#3}}%
}
\newcommand*{\Family@@String@@Key}{}
\long\def\Family@@Numerical@@Key#1#2#3[#4]#5{%
  \DefineFamilyKey[{#1}]{#2}{#3}[{#4}]{\def#5{##1}}%
}
\newcommand{\Family@String@Key}[4]{%
  \DefineFamilyKey[{#1}]{#2}{#3}{\def#4{##1}}%
}
⟨/base⟩
```

\KOMA@DeclareDeprecatedOption
\KOMA@@DeclareDeprecatedOption
\KOMA@@@DeclareDeprecatedOption

Mit Hilfe dieses Makros kann eine Option definiert werden, die eigentlich obsolete ist. Statt dieser Option wird dann ein Schlüssel verwendet. Das erste Argument ist optional und muss dann der Paketname sein. Ist es nicht gesetzt, so wird stattdessen die Klasse `\KOMAClassName` angenommen. Das zweite Argument ist der Name der Option. Das vierte Argument ist das Setzen des gewünschten Schlüssels.

```
⟨*koma⟩
\newcommand*{\KOMA@DeclareDeprecatedOption}{%
```

```
  \@ifnextchar [%]
  {\KOMA@@DeclareDeprecatedOption}%
  {\KOMA@@@DeclareDeprecatedOption{\ClassWarningNoLine{\KOMAClassName}}}%
}
\newcommand*{\KOMA@@DeclareDeprecatedOption}[1][]{%
  \KOMA@@@DeclareDeprecatedOption{\PackageWarningNoLine{#1}}%
}
\newcommand*{\KOMA@@@DeclareDeprecatedOption}[3]{%
  \DeclareOption{#2}{\KOMA@UseDeprecatedOption{#1}{#2}{#3}}%
}
```

`\KOMA@UseDeprecatedOption` Die eigentliche Arbeit wird von diesem Makro erledigt. Die Argumente sind 1. Meldungsbefehl, 2. obsolete Option, 3. Setzen des Schlüssels.

```
\newcommand*{\KOMA@UseDeprecatedOption}[3]{%
  \let\scr@sr@co\CurrentOption
  #1{%
    You've used obsolete option `#2'.\MessageBreak
    Usage of this option indicates an old document and\MessageBreak
    changes compatibility level using\MessageBreak
    `#3,version=first'.\MessageBreak
    If you don't want this, you should simply replace\MessageBreak
    usage of option `#2' by `#3'%
  }%
  \KOMAExecuteOptions{#3,version=first}%
  \let\CurrentOption\scr@sr@co
}
⟨/koma⟩
```

`\KOMA@DeclareStandardOption`
`\KOMA@@DeclareStandardOption`
`\KOMA@@@DeclareStandardOption`

Mit Hilfe dieses Makros kann eine Option definiert werden, die eigentlich obsolete ist. Statt dieser Option wird dann ein Schlüssel verwendet. Das erste Argument ist optional und muss dann der Paketname sein. Ist es nicht gesetzt, so wird stattdessen die Klasse `\KOMAClassName` angenommen. Das zweite Argument ist der Name der Option. Das vierte Argument ist das Setzen des gewünschten Schlüssels.

```
⟨*koma⟩
\newcommand*{\KOMA@DeclareStandardOption}{%
  \@ifnextchar [%]
  {\KOMA@@DeclareStandardOption}%
  {\KOMA@@@DeclareStandardOption{\ClassInfoNoLine{\KOMAClassName}}}%
}
\newcommand*{\KOMA@@DeclareStandardOption}[1][]{%
  \KOMA@@@DeclareStandardOption{\PackageInfoNoLine{#1}}%
}
\newcommand*{\KOMA@@@DeclareStandardOption}[3]{%
  \DeclareOption{#2}{\KOMA@UseStandardOption{#1}{#2}{#3}}%
}
```

\KOMA@UseStandardOption

Die eigentliche Arbeit wird von diesem Makro erledigt. Die Argumente sind 1. Meldungsbefehl, 2. obsolete Option, 3. Setzen des Schlüssels.

```
\newcommand*{\KOMA@UseStandardOption}[3]{%
  \let\scr@sr@co\CurrentOption
  #1{%
    You've used standard option `#2'.\MessageBreak
    This is correct!\MessageBreak
    Internally I'm using `#3'.\MessageBreak
    If you'd like to set the option with \string\KOMAoptions,\MessageBreak
    you'd have to use `#3' there\MessageBreak
    instead of `#2', too%
  }%
  \KOMAExecuteOptions{#3}%
  \let\CurrentOption\scr@sr@co
}
⟨/koma⟩
```

\KOMA@DeclareObsoleteOption
\KOMA@@DeclareObsoleteOption
\KOMA@@@DeclareObsoleteOption
\KOMA@iv@DeclareObsoleteOption

Mit Hilfe dieses Makros kann eine Option definiert werden, die eigentlich obsolete ist. Statt dieser Option wird dann ein Schlüssel verwendet. Das erste Argument ist optional und muss dann entweder `\PackageInfo`, `\PackageInfoNoLine`, `\PackageWarning`, `\PackageWarningNoLine` oder das gleiche für Klassen sein. Das zweite Argument ist ebenfalls optional und der Name des Pakets oder der Klasse. Das dritte Argument ist der Name der Option. Das vierte Argument ist das Setzen des gewünschten Schlüssels.

```
⟨*koma⟩
\newcommand*{\KOMA@DeclareObsoleteOption}{%
  \@ifnextchar [%]
  {\KOMA@@DeclareObsoleteOption}%
  {\KOMA@@DeclareObsoleteOption[\@gobbletwo]}%
}
\newcommand*{\KOMA@@DeclareObsoleteOption}{}
\def\KOMA@@DeclareObsoleteOption[#1]{%
  \@ifnextchar [%]
  {\KOMA@@@DeclareObsoleteOption{#1}}%
  {\KOMA@@@DeclareObsoleteOption{#1}[]}%
}
\newcommand*{\KOMA@@@DeclareObsoleteOption}{}
\def\KOMA@@@DeclareObsoleteOption#1[#2]#3#4{%
  \ifstr{#2}{}{%
    \KOMA@iv@DeclareObsoleteOption{#1}{\@currname}{#3}{#4}%
  }{%
    \KOMA@UseObsoleteOption{#1}{#2}{#3}{#4}%
  }%
}
\newcommand*{\KOMA@iv@DeclareObsoleteOption}[4]{%
  \DeclareOption{#3}{\KOMA@UseObsoleteOption{#1{#2}}{#3}{#4}}%
}
```

\KOMA@UseObsoleteOption Die eigentliche Arbeit wird von diesem Makro erledigt. Die Argumente sind 1. Meldungsbefehl, 2. obsolete Option, 3. Setzen des Schlüssels.

```
\newcommand*{\KOMA@UseObsoleteOption}[3]{%
  \let\scr@sr@co\CurrentOption
  #1{%
    You've used obsolete option `#2'.\MessageBreak
    \protect\KOMAExecuteOptions{#3} will be\MessageBreak
    used instead.\MessageBreak
    You may simply replace `#2'\MessageBreak
    by `#3'%
  }%
  \KOMAExecuteOptions{#3}%
  \let\CurrentOption\scr@sr@co
}
```

\scr@sr@co Da das xkeyval-Paket leider \CurrentOption vernichtet, das aber innerhalb von \ProcessOptions und \KOMAProcessOptions nach dem Ausführen der Option noch benötigt wird, muss die Anweisung gesichert werden. Dafür wird ein Hilfsmakro benötigt.

```
\newcommand*{\scr@sr@co}{}
⟨/koma⟩
```

\KOMA@newobsoletecommand Mit Hilfe dieses Makros kann ein Makro definiert werden, das eigentlich obsolete ist. Das Makro verwendet dann schlicht einen Schlüssel. Das erste Argument ist optional und muss dann entweder \PackageInfo, \PackageInfoNoLine, \PackageWarning oder \PackageWarningNoLine sein. Das zweite Argument ist das Makro. Das dritte Argument ist das Setzen des gewünschten Schlüssels. Das definierte Makro selbst hat Schaltercharakter, ist also nicht \long und versteht auch keine Argumente.

```
⟨*koma⟩
\newcommand*{\KOMA@newobsoletecommand}[3][\@gobbletwo]{%
  \newcommand*{#2}{\KOMA@UseObsoleteCommand{#1}{\string#2}{#3}}%
}
```

\KOMA@UseObsoleteCommand Die eigentliche Arbeit wird von diesem Makro erledigt. Die Argumente sind dieselben.

```
\newcommand*{\KOMA@UseObsoleteCommand}[3]{%
  #1{%
    scrkbase%
  }{%
    You've used the obsolete command `#2'.\MessageBreak
    \protect\KOMAoptions{#3} will be used instead.\MessageBreak
    You should also replace `#2' by `\protect\KOMAoptions{#3}'%
  }%
  \KOMAoptions{#3}%
}
⟨/koma⟩
```

Datei c
scrkmisc.dtx

3 Verschiedenes

Es gibt eine ganze Menge für das sich einzeln keine eigene Datei lohnt oder eine Einordnunge schwierig ist. All das nennen wird „Verschiedenes“.

3.1 Erweiterung der Internationalisierung

Die Standardnamen wie `\figurename` etc. sind nicht immer ausreichend, da manche Sprachen Zählwörter vor oder nach dem jeweiligen Zähler einfügen müssen (man denke da sogar im Deutschen an „Abbildung 1“ und „Erste Abbildung“). Desweiteren werden bei manchen Sprachen die Zwischenräume anders gesetzt bzw. sogar ganz unterdrückt.

Es gibt drei Ebenen, die man modifizieren kann, um verschiedenste Sprachen zu implementieren: Ebene 1 besteht aus den hinlänglich bekannten Makros wie `\figurename` und alle weiteren Makros, die z. B. `german` modifiziert. Desweiteren können die Makros `\sectfont` etc. etwaige zusätzliche Fonts aktivieren. Ebene 2 ist die zusätzliche Adaptierung von den Zählern selbst (`\thesection` könnte z. B. chinesische Schriftzeichen ansprechen anstelle von arabischen Ziffern). Die allgemeinste Form der dritten Ebene besteht aus den Makros `\chaptermarkformat`, `\sectionmarkformat` etc. welche die Änderung der Reihenfolge von Zähler und Titel sowie die genaue Kontrolle der Zwischenräume etc. ermöglichen.

Für die gängigen Sprachen können alle Makros für die dritte Ebene identisch definiert werden. Bei Verwendung von „exotischen“ Sprachen, aber auch speziellen Darstellungen in den gängigen Sprachen können diese Makros umdefiniert werden.

Das Konzept für der Internationalisierung mit der dritten Ebene wurde ursprünglich von Werner Lemberg[1] entwickelt. Die „caption files“ seines CJK-Pakets (**C**hinesisch, **J**apanisch, **K**oreanisch) bauen auf diesem Konzept auf. Damit ist KOMA-Script für alle möglichen Sprachanpassungen bestens gerüstet. In der dritten Ebene wird übrigens auch die Einhaltung der DUDEN-Regeln für die Gliederungsnummern realisiert.

3.2 Optionen für Entwürfe

`draft` `final`

Der einzige Unterschied zwischen Entwürfen und Endversionen ist aus Sicht der Klassen, ob Zeilenüberhänge mit einem schwarzen Balken angezeigt werden sollen oder nicht.

[1] Werner Lemberg ist im InterNet unter `a7621gac@awiuni11.bitnet` zu erreichen.

```
\KOMA@key{draft}[true]{%
  \KOMA@set@ifkey{draft}{@tempswa}{#1}%
  \if@tempswa
    \setlength{\overfullrule}{5pt}%
  \else
    \setlength{\overfullrule}{\z@}%
  \fi
}
\KOMA@DeclareStandardOption
⟨package & extend⟩[scrextend]
{final}{draft=false}
```

3.3 Optionen für Gleichungen

Briefe haben derzeit keine Gleichungen.

`leqno` `fleqno` Es können Gleichungsnummern links oder linksbündige Gleichungen eingestellt werden.

```
⟨*!extend⟩
\DeclareOption{leqno}{\input{leqno.clo}}
\DeclareOption{fleqn}{\input{fleqn.clo}}
⟨/!extend⟩
```

3.4 Standardeinstellungen für die Optionen

Am Ende aller Optionen wird zunächst die Standardeinstellungen der Optionen vorgenommen und dann alle Optionen ausgeführt. Dies geschieht zwischen den Optionen und dem Körper dieser Datei.

Es müssen die Standardoptionen festgelegt und alle Optionen ausgewählt werden.

```
⟨*!extend⟩
\KOMAExecuteOptions{%
  parskip=false,%
  headings=big%
}
⟨/!extend⟩
\KOMAProcessOptions\relax
```

3.5 Temporäre Umschaltung zwischen ein- und zweispaltigem Satz

Briefe sind immer einspaltig, benötigen also den Schalter nicht.

`\if@restonecol` `\@restonecoltrue` `\@restonecolfalse` Damit innerhalb einer Umgebung lokal auf ein- oder zweispaltigen Satz umgeschaltet werden kann, wird ein Schalter benötigt. Dieser wird dann dafür verwendet, sich zu merken, dass eine Umschaltung stattgefunden hat und am Ende der Umgebung wieder aufzuheben ist. Der Schalter wird nicht

in Umgebungen verwendet, die geschachtelt werden. Verwendung findet er beispielsweise für Titelseiten und das Inhaltsverzeichnis.

```
⟨*!extend⟩
\newif\if@restonecol
⟨/!extend⟩
⟨*extend⟩
\scr@ifundefinedorrelax{if@restonecol}{%
  \expandafter\newif\csname if@restonecol\endcsname
  \@restonecolfalse
}{}
⟨/extend⟩
```

3.6 Definitionen für den schlauen Spruch

Den schlauen Spruch gibt es nicht bei Briefen.

`\dictum`
`\dictumwidth`
`\raggeddictum`
`\raggeddictumtext`
`\raggeddictumauthor`
`dictum`
`dictumtext`
`dictumauthor`
`\dictumauthorformat`

Ein *dictum* ist ein Spruch. Er besteht aus dem Spruch selbst und verfügt optional über einen Autor, der dann mit einer Linie abgetrennt wird. Gibt es keinen Autor, gibt es auch keine Linie. Die Schriften für Autor und Spruch können getrennt voneinander eingestellt werden. Die Anordnung des Spruchs wird durch `\raggeddictumtext` gesetzt. Die des Autors durch `\raggeddictumauthor`. Die Formatierung selbst geschieht mit dem Makro `\dictumauthorformat`. Vordefiniert ist, den Namen des Autors in runde Klammern zu setzen. Die gesamte Geschichte von Text und Autor wird in eine Box der Breite `\dictumwidth` gepackt. Diese wird mit `\raggeddictum` angeordnet.

```
\newcommand{\dictum}[2][]{\par%
  \begingroup
    \raggeddictum\parbox{\dictumwidth}{%
      {\@dictumfont{\raggeddictumtext #2\strut\par}%
        \def\@tempa{#1}\ifx\@tempa\@empty\else%
          {\raggeddictumtext\vskip-1ex\hrulefill\par}%
          \raggeddictumauthor\@dictumauthorfont\dictumauthorformat{#1}%
          \strut\par%
        \fi%
      }%
    }\par%
  \endgroup
}
\newcommand*{\dictumwidth}{.3333\textwidth}
\newcommand*{\raggeddictum}{\raggedleft}
\newcommand*{\dictumauthorformat}[1]{(#1)}
\newcommand*{\raggeddictumtext}{\raggedright}
\newcommand*{\raggeddictumauthor}{\raggedleft}
\newkomafont{dictum}{\normalfont\normalcolor\sffamily\small}
\newkomafont{dictumauthor}{\itshape}
\aliaskomafont{dictumtext}{dictum}
```

3.7 Längen für Minipages und Rahmen

@mpfootins
fboxsep
fboxrule

\@mpfootins entspricht \footins in Minipages. \fboxsep ist der Abstand zwischen dem Rahmen einer \fbox oder \framebox und dem Inhalt, \fboxrule ist die Dicke des Rahmens.

```
⟨*!extend⟩
\setlength{\skip\@mpfootins}{\skip\footins}
\setlength\fboxsep{3\p@}
\setlength\fboxrule{.4\p@}
⟨/!extend⟩
```

3.8 Gleichungen

Gleichungen gibt es bei Briefen nicht.

\theequation

Darstellung der Gleichungsnummer:

```
⟨*!extend⟩
⟨book | report⟩\@addtoreset{equation}{chapter}
\renewcommand*\theequation{%
⟨*book⟩
  \ifnum \scr@compatibility>\@nameuse{scr@v@2.97d}\relax
    \if@mainmatter\thechapter.\fi
  \else
⟨/book⟩
⟨book | report⟩   \thechapter.%
⟨book⟩   \fi
  \@arabic\c@equation
}
⟨/!extend⟩
```

3.9 Definitionen zur Feststellung von linken und rechten Seiten

\ifthispageodd

Häufig wird mit \ifodd\value{page} getestet, ob man sich gerade auf einer rechten oder einer linken Seite befindet. Dummerweise funktioniert das aber genaugenommen nur zufällig manchmal. Ob man sich auf einer linken oder einer rechten Seite befindet, ergibt sich nämlich erst bei der Ausgabe. Deshalb wird hier ein Makro definiert, das über eine Art Label-Mechanismus arbeitet. Es werden bewusst keine echten \label und \pageref verwendet, weil diese Befehle von verschiedenen Paketen manipuliert werden.

\new@tpo@label

Definition eines neuen *true-page-odd* Labels. Wird in der aux-Datei verwendet.

```
\newcommand*{\new@tpo@label}{\@newl@bel{tpo}}
```

\scr@tpo

Es wird ein Zähler benötigt, der die aktuelle Nummer der Anfrage beinhaltet.

```
\newcommand*{\scr@tpo}{0}
```

`\is@thispageodd` Dieses Makro macht die eigentliche Arbeit. Es funktioniert nicht wie ein normales `\iffalse` oder `\iftrue`, denn es kann nicht geschachtelt innerhalb anderer `\if... ...\fi` Klammerungen verwendet werden. Es sollte daher niemals außerhalb von `\ifthispageodd` verwendet werden. Da über die `aux`-Datei gearbeitet wird, funktioniert das Ganze nur, wenn die Datei auch geschrieben wird und min. zwei LaTeX-Läufe erfolgen.

```
\newcommand*{\is@thispageodd}{%
  \@bsphack
  \begingroup
    \@tempcnta=\scr@tpo
    \advance\@tempcnta by\@ne
    \xdef\scr@tpo{\the\@tempcnta}%
    \protected@write\@auxout{\let\arabic\relax}{%
      \string\new@tpo@label{\scr@tpo}{\arabic{page}}}%
    \expandafter\ifx\csname tpo@\scr@tpo\endcsname\relax
      \protect\G@refundefinedtrue
⟨!extend⟩       \ClassWarning{\KOMAClassName}{%
⟨extend⟩       \PackageWarning{scrextend}{%
         odd/even page label number \scr@tpo\space undefined}%
      \edef\@tempa{\the\value{page}}%
    \else
      \edef\@tempa{\csname tpo@\scr@tpo\endcsname}%
    \fi
    \ifodd\number\@tempa
      \aftergroup\thispagewasoddtrue
    \else
      \aftergroup\thispagewasoddfalse
    \fi
  \endgroup
  \@esphack
}
```

`\ifthispagewasodd` `\thispagewasoddtrue` `\thispagewasoddfalse` Dieser Schalter speichert auf Rolf Niepraschks speziellen Wunsch die letzte Entscheidung von `\ifthispageodd`. Vor dem ersten `\ifthispageodd` ist der Zustand undefiniert. Es ist auch nicht garantiert, dass die Entscheidung noch zutreffend ist.

```
\newif\ifthispagewasodd
```

Nun fehlt nur noch das eigentliche oben versprochene Anwendermacro. Das ist nun aber auch kein Problem mehr:

```
\newcommand{\ifthispageodd}{%
  \is@thispageodd
  \ifthispagewasodd
    \expandafter\@firstoftwo
  \else
    \expandafter\@secondoftwo
  \fi
}
```

3.10 Definitionen für tiefer gestellten Text

`textsubscript`
`@textsubscript`

Im LaTeX-Kern wird zwar ein Makro `\textsuperscript` definiert, um Texte hochstellen zu können, die entsprechende Anweisung zum Tiefstellen fehlt jedoch. Das wird hier nachgeholt, wobei nahezu unverändert auf die Definition aus dem LaTeX-Kern zurückgegriffen wird.

```
\DeclareRobustCommand*\textsubscript[1]{%
  \@textsubscript{\selectfont#1}%
}
\newcommand{\@textsubscript}[1]{%
  {\m@th\ensuremath{_{\mbox{\fontsize\sf@size\z@#1}}}}%
}
```

Datei d scrklang.dtx

4 Sprachabhängigkeiten

Aus verschiedenen Gründen ist es nicht so einfach, neue, sprachabhängige Begriffe zu definieren oder vorhandene umzudefinieren. Mit KOMA-Script geht das etwas einfacher.

4.1 Option

Für Briefe gibt es die Möglichkeit, zwischen symbolischem und numerischem Datum umzuschalten.

`numericaldate` Es kann zwischen dem Originaldatum, das beispielsweise von `german.sty` definiert wird, und dem `scrlttr2`-Datum gewählt werden. Natürlich gibt es diese Optionen nur bei Briefen.

`\if@orgdate` `\@orgdatetrue` `\@orgdatefalse` Die Einstellung wird in einem Schalter gespeichert.

```
\newif\if@orgdate\@orgdatetrue
```

Dummerweise besitzt der alte Schalter genau die gegenteilige Bedeutung der Option, deshalb muss der Schalter zweimal invertiert werden.

```
\KOMA@key{numericaldate}[true]{%
  \if@orgdate\@orgdatefalse\else\@orgdatetrue\fi
  \KOMA@set@ifkey{numericaldate}{@orgdate}{#1}%
  \if@orgdate\@orgdatefalse\else\@orgdatetrue\fi
}
```

`scrdate` `orgdate`

```
\KOMA@DeclareDeprecatedOption{scrdate}{numericaldate=true}
\KOMA@DeclareDeprecatedOption{orgdate}{numericaldate=false}
```

4.2 Definitionen für sprachabhängige Bezeichner

Dieser Teil ist so grundlegend, dass er ab Version 3.00 in das Paket scrbase ausgelagert wird.

`\providecaptionname` `\newcaptionname` `\renewcaptionname` Drei Benutzerkommandos zur Definition sprachabhängiger Begriffe. Es empfiehlt sich wegen babel diese Anweisungen innerhalb von `\AtBeginDocument` zu verwenden.

```
\newcommand*{\providecaptionname}[3]{%
  \@ifundefined{captions#1}{%
    \@ifundefined{cap@info@#1@done}{%
      \PackageInfo{scrbase}{%
```

```
        No captions found for ‘#1’\MessageBreak
        --> skipped%
      }%
      \expandafter\let\csname cap@info@#1@done\endcsname\@empty
    }{}%
  }{%
    \begingroup
      \let\@tempa\endgroup
      \def\@tempb{\relax\undefined\relax}%
      \let #2\@tempb\relax
      \csname captions#1\endcsname
      \ifx #2\@tempb\relax
        \edef\@tempb{#1}%
        \@onelevel@sanitize\@tempb
        \@onelevel@sanitize\languagename
        \ifx\@tempb\languagename
          \def\@tempa{\endgroup%
            \expandafter\g@addto@macro\csname captions#1\endcsname{%
              \def#2{#3}}%
            \csname captions#1\endcsname
          }%
        \else
          \def\@tempa{\endgroup%
            \expandafter\g@addto@macro\csname captions#1\endcsname{%
              \def#2{#3}}%
          }%
        \fi
      \else
        \PackageInfo{scrbase}{%
          ‘\string#2’ at language ‘#1’ already
          defined\MessageBreak
          --> skipped%
        }%
      \fi
    \@tempa
  }%
}
\newcommand*{\newcaptionname}[3]{%
  \@ifundefined{captions#1}{%
    \PackageInfo{scrbase}{%
      No captions found for ‘#1’ --> defining%
    }%
    \@namedef{captions#1}{}%
  }{}%
  \begingroup
    \def\@tempb{\relax\undefined\relax}%
    \let #2\@tempb\relax
    \csname captions#1\endcsname
    \ifx #2\@tempb\relax\else
```

```
      \PackageError{scrbase}{%
        `\string#2' at language `#1' already defined%
      }{%
        You've told me to define `\string#2' at language
        `#1',\MessageBreak
        but it is already defined.\MessageBreak
        It will be redefined, if you'll continue.%
      }%
    \fi
  \endgroup
  \expandafter\g@addto@macro\csname captions#1\endcsname{%
    \def#2{#3}%
  }%
  \begingroup
    \edef\@tempb{#1}%
    \@onelevel@sanitize\@tempb
    \@onelevel@sanitize\languagename
    \ifx\@tempb\languagename
      \def\@tempa{\endgroup\csname captions#1\endcsname}%
    \else
      \let\@tempa\endgroup
    \fi
  \@tempa
}
\newcommand*{\renewcaptionname}[3]{%
  \@ifundefined{captions#1}{%
    \PackageError{scrbase}{%
      No captions found for `#1'%
    }{%
      You've told me to redefine the caption `\string#2' at language
      `#1',\MessageBreak
      but `\expandafter\string\csname captions#1\endcsname' is not
      defined.\MessageBreak
      If you'll continue it will be defined.%
    }
    \@namedef{captions#1}{}%
  }{}%
  \begingroup
    \def\@tempb{\relax\undefined\relax}%
    \let #2\@tempb\relax
    \csname captions#1\endcsname
    \ifx #2\@tempb\relax
      \PackageError{scrbase}{%
        `\string#2' not defined at language `#1'%
      }{%
        You've told me to redefine `\string#2' at language
        `#1',\MessageBreak
        but it is not defined.\MessageBreak
        If you'll continue it will be defined.%
```

```
      }%
    \fi
  \endgroup
  \expandafter\g@addto@macro\csname captions#1\endcsname{%
    \def#2{#3}%
  }%
  \begingroup
    \edef\@tempb{#1}%
    \@onelevel@sanitize\@tempb
    \@onelevel@sanitize\languagename
    \ifx\@tempb\languagename
      \def\@tempa{\endgroup\csname captions#1\endcsname}%
    \else
      \let\@tempa\endgroup
    \fi
  \@tempa
}
```

4.3 Definitionen für ein nummerisches Datum

Bei Briefen kann ein nummerisches oder ein symbolisches Datum verwendet werden.

`\g@addnumerical@date` Dazu ist es notwendig, ggf. das bereits definierte symbolische Datum durch ein nummerisches zu ergänzen.

```
\newcommand*{\g@addnumerical@date}[2]{%
  \@ifundefined{date#1}{%
    \ClassInfo{%
      scrlttr2%
    }{%
      no date found for language ‘#1’\MessageBreak
      --> skipped%
    }%
  }{%
    \expandafter\g@addto@macro\csname date#1\endcsname{%
      \let\sym@date=\today%
      \def\num@date{#2}%
      \def\today{\if@orgdate\sym@date\else\num@date\fi}%
    }%
  }%
}
```

4.4 Festlegungen für einzelne Sprachen

Es folgen nun für einzelne Sprachen konkrete Festlegungen, die für Briefe benötigt werden.

`\captionsenglish` `\dateenglish` `\captionsUSenglish` `\dateUSenglish` `\captionsamerican` `\dateamerican` `\captionsbritish` `\datebritish` `\captionsUKenglish` `\dateUKenglish` `\captionsgerman` `\dategerman` `\captionsngerman` `\datengerman`

Weil es in früheren Versionen zu Problemen damit gekommen ist, werden diese jedoch erst bei `\begin{document}` definiert. Dabei wird `\providecaptionname`

Sprache	Spender
English	Frank Neukam, Michael Dewey, Markus Kohm
Deutsch	Frank Neukam, Markus Kohm
Französisch	Frank Neukam, Henk Jongbloets
Italienisch	Simone Naldi
Spanisch	Ralph J. Hangleiter, Alejandro López-Valencia
Kroatisch	Branka Lončarević
Holländisch	Henk Jongbloets
Finnisch	Hannu Väisänen
Norwegisch	Sveinung Heggen
Schwedisch	Benjamin Hell

Tabelle 1: Liste der unterstützten Sprachen und der *Sprachspender*

verwendet, so dass sie auch schon vor `\begin{document}` mit `\newcaptionname` anderslautend definiert werden können. Eine Besonderheit stellt die englische Sprache dar. Es wird versucht, diese als Notersatz immer zu definieren.

```
\AtBeginDocument{%
  \@ifundefined{captionsenglish}{\let\captionsenglish\@empty}{}%
  \@ifundefined{dateenglish}{\def\dateenglish{%
      \def\today{\ifcase\month\or
        January\or February\or March\or April\or May\or June\or
        July\or August\or September\or October\or November\or December\fi
        \space\number\day, \number\year}%
    }%
  }{}%
  \providecaptionname{english}\yourrefname{Your ref.}%
  \providecaptionname{english}\yourmailname{Your letter of}%
  \providecaptionname{english}\myrefname{Our ref.}%
  \providecaptionname{english}\customername{Customer no.}%
  \providecaptionname{english}\invoicename{Invoice no.}%
  \providecaptionname{english}\subjectname{Subject}%
  \providecaptionname{english}\ccname{cc}%
  \providecaptionname{english}\enclname{encl}%
  \providecaptionname{english}\headtoname{To}%
  \providecaptionname{english}\headfromname{From}%
  \providecaptionname{english}\datename{Date}%
  \providecaptionname{english}\pagename{Page}%
  \providecaptionname{english}\phonename{Phone}%
  \providecaptionname{english}\faxname{Fax}%
  \providecaptionname{english}\emailname{Email}%
  \providecaptionname{english}\wwwname{Url}%
  \providecaptionname{english}\bankname{Bank account}%
  \g@addnumerical@date{english}{\number\day/\number\month/\number\year}%
  \providecaptionname{USenglish}\yourrefname{Your ref.}%
```

```
\providecaptionname{USenglish}\yourmailname{Your letter of}%
\providecaptionname{USenglish}\myrefname{Our ref.}%
\providecaptionname{USenglish}\customername{Customer no.}%
\providecaptionname{USenglish}\invoicename{Invoice no.}%
\providecaptionname{USenglish}\subjectname{Subject}%
\providecaptionname{USenglish}\ccname{cc}%
\providecaptionname{USenglish}\enclname{encl}%
\providecaptionname{USenglish}\headtoname{To}%
\providecaptionname{USenglish}\headfromname{From}%
\providecaptionname{USenglish}\datename{Date}%
\providecaptionname{USenglish}\pagename{Page}%
\providecaptionname{USenglish}\phonename{Phone}%
\providecaptionname{USenglish}\faxname{Fax}%
\providecaptionname{USenglish}\emailname{Email}%
\providecaptionname{USenglish}\wwwname{Url}%
\providecaptionname{USenglish}\bankname{Bank account}%
\g@addnumerical@date{USenglish}{\number\month/\number\day/\number\year}%
\providecaptionname{american}\yourrefname{Your ref.}%
\providecaptionname{american}\yourmailname{Your letter of}%
\providecaptionname{american}\myrefname{Our ref.}%
\providecaptionname{american}\customername{Customer no.}%
\providecaptionname{american}\invoicename{Invoice no.}%
\providecaptionname{american}\subjectname{Subject}%
\providecaptionname{american}\ccname{cc}%
\providecaptionname{american}\enclname{encl}%
\providecaptionname{american}\headtoname{To}%
\providecaptionname{american}\headfromname{From}%
\providecaptionname{american}\datename{Date}%
\providecaptionname{american}\pagename{Page}%
\providecaptionname{american}\phonename{Phone}%
\providecaptionname{american}\faxname{Fax}%
\providecaptionname{american}\emailname{Email}%
\providecaptionname{american}\wwwname{Url}%
\providecaptionname{american}\bankname{Bank account}%
\g@addnumerical@date{american}{\number\month/\number\day/\number\year}%
\providecaptionname{british}\yourrefname{Your ref.}%
\providecaptionname{british}\yourmailname{Your letter of}%
\providecaptionname{british}\myrefname{Our ref.}%
\providecaptionname{british}\customername{Customer no.}%
\providecaptionname{british}\invoicename{Invoice no.}%
\providecaptionname{british}\subjectname{Subject}%
\providecaptionname{british}\ccname{cc}%
\providecaptionname{british}\enclname{encl}%
\providecaptionname{british}\headtoname{To}%
\providecaptionname{british}\headfromname{From}%
\providecaptionname{british}\datename{Date}%
\providecaptionname{british}\pagename{Page}%
\providecaptionname{british}\phonename{Phone}%
\providecaptionname{british}\faxname{Fax}%
```

```
\providecaptionname{british}\emailname{Email}%
\providecaptionname{british}\wwwname{Url}%
\providecaptionname{british}\bankname{Bank account}%
\g@addnumerical@date{british}{\number\day/\number\month/\number\year}%
\providecaptionname{UKenglish}\yourrefname{Your ref.}%
\providecaptionname{UKenglish}\yourmailname{Your letter of}%
\providecaptionname{UKenglish}\myrefname{Our ref.}%
\providecaptionname{UKenglish}\customername{Customer no.}%
\providecaptionname{UKenglish}\invoicename{Invoice no.}%
\providecaptionname{UKenglish}\subjectname{Subject}%
\providecaptionname{UKenglish}\ccname{cc}%
\providecaptionname{UKenglish}\enclname{encl}%
\providecaptionname{UKenglish}\headtoname{To}%
\providecaptionname{UKenglish}\headfromname{From}%
\providecaptionname{UKenglish}\datename{Date}%
\providecaptionname{UKenglish}\pagename{Page}%
\providecaptionname{UKenglish}\phonename{Phone}%
\providecaptionname{UKenglish}\faxname{Fax}%
\providecaptionname{UKenglish}\emailname{Email}%
\providecaptionname{UKenglish}\wwwname{Url}%
\providecaptionname{UKenglish}\bankname{Bank account}%
\g@addnumerical@date{UKenglish}{\number\day/\number\month/\number\year}%
\providecaptionname{german}\yourrefname{Ihr Zeichen}%
\providecaptionname{german}\yourmailname{Ihr Schreiben vom}%
\providecaptionname{german}\myrefname{Unser Zeichen}%
\providecaptionname{german}\customername{Kundennummer}%
\providecaptionname{german}\invoicename{Rechnungsnummer}%
\providecaptionname{german}\subjectname{Betrifft}%
\providecaptionname{german}\ccname{Kopien an}%
\providecaptionname{german}\enclname{Anlagen}%
\providecaptionname{german}\headtoname{An}%
\providecaptionname{german}\headfromname{Von}%
\providecaptionname{german}\datename{Datum}%
\providecaptionname{german}\pagename{Seite}%
\providecaptionname{german}\phonename{Telephon}%
\providecaptionname{german}\faxname{Fax}%
\providecaptionname{german}\emailname{E-Mail}%
\providecaptionname{german}\wwwname{URL}%
\providecaptionname{german}\bankname{Bankverbindung}%
\g@addnumerical@date{german}{\number\day.\,\number\month.\,\number\year}%
\providecaptionname{ngerman}\yourrefname{Ihr Zeichen}%
\providecaptionname{ngerman}\yourmailname{Ihr Schreiben vom}%
\providecaptionname{ngerman}\myrefname{Unser Zeichen}%
\providecaptionname{ngerman}\customername{Kundennummer}%
\providecaptionname{ngerman}\invoicename{Rechnungsnummer}%
\providecaptionname{ngerman}\subjectname{Betrifft}%
\providecaptionname{ngerman}\ccname{Kopien an}%
\providecaptionname{ngerman}\enclname{Anlagen}%
\providecaptionname{ngerman}\headtoname{An}%
```

```
\providecaptionname{ngerman}\headfromname{Von}%
\providecaptionname{ngerman}\datename{Datum}%
\providecaptionname{ngerman}\pagename{Seite}%
\providecaptionname{ngerman}\phonename{Telefon}%
\providecaptionname{ngerman}\faxname{Fax}%
\providecaptionname{ngerman}\emailname{E-Mail}%
\providecaptionname{ngerman}\wwwname{URL}%
\providecaptionname{ngerman}\bankname{Bankverbindung}%
\g@addnumerical@date{ngerman}{\number\day.\,\number\month.\,\number\year}%
\providecaptionname{austrian}\yourrefname{Ihr Zeichen}%
\providecaptionname{austrian}\yourmailname{Ihr Schreiben vom}%
\providecaptionname{austrian}\myrefname{Unser Zeichen}%
\providecaptionname{austrian}\customername{Kundennummer}%
\providecaptionname{austrian}\invoicename{Rechnungsnummer}%
\providecaptionname{austrian}\subjectname{Betrifft}%
\providecaptionname{austrian}\ccname{Kopien an}%
\providecaptionname{austrian}\enclname{Anlagen}%
\providecaptionname{austrian}\headtoname{An}%
\providecaptionname{austrian}\headfromname{Von}%
\providecaptionname{austrian}\datename{Datum}%
\providecaptionname{austrian}\pagename{Seite}%
\providecaptionname{austrian}\phonename{Telefon}%
\providecaptionname{austrian}\faxname{Fax}%
\providecaptionname{austrian}\emailname{E-Mail}%
\providecaptionname{austrian}\wwwname{URL}%
\providecaptionname{austrian}\bankname{Bankverbindung}%
\g@addnumerical@date{austrian}{\number\day.\,\number\month.\,\number\year}%
\providecaptionname{french}\yourrefname{Vos r\'ef\'erences}%
\providecaptionname{french}\yourmailname{Votre lettre du}%
\providecaptionname{french}\myrefname{Nos r\'ef\'erences}%
\providecaptionname{french}\customername{Num\'ero de client}%
\providecaptionname{french}\invoicename{Num\'ero de facture}%
\providecaptionname{french}\subjectname{Concernant}%
\providecaptionname{french}\ccname{Copie \`a}%
\providecaptionname{french}\enclname{Annexes}%
\providecaptionname{french}\headtoname{A}%
\providecaptionname{french}\headfromname{De}%
\providecaptionname{french}\datename{Date}%
\providecaptionname{french}\pagename{Page}%
\providecaptionname{french}\phonename{T\'el\'ephone}%
\providecaptionname{french}\faxname{T\'el\'efax}%
\providecaptionname{french}\emailname{E-mail}%
\providecaptionname{french}\wwwname{URL}%
\providecaptionname{french}\bankname{Compte en banque}%
\g@addnumerical@date{french}{\number\day.\,\number\month.\,\number\year}%
\providecaptionname{italian}\yourrefname{Vs./Rif.}% or Vostro Riferimento
\providecaptionname{italian}\yourmailname{Vs.~lettera del}% or Vostra
                                                          % lettera del
\providecaptionname{italian}\myrefname{Ns./Rif.}% or Nostro Riferimento
```

```
\providecaptionname{italian}\customername{Nr.~cliente}% or Cliente num.
\providecaptionname{italian}\invoicename{Nr.~fattura}% or Fattura num.
\providecaptionname{italian}\subjectname{Oggetto}%
\providecaptionname{italian}\ccname{Per conoscenza}% or Copia a
\providecaptionname{italian}\enclname{Allegato}% or (plural) Allegati
\providecaptionname{italian}\headtoname{A}%
\providecaptionname{italian}\headfromname{Da}%
\providecaptionname{italian}\datename{Data}%
\providecaptionname{italian}\pagename{Pagina}%
\providecaptionname{italian}\phonename{Telefono}%
\providecaptionname{italian}\faxname{Fax}%
\providecaptionname{italian}\emailname{Email}%
\providecaptionname{italian}\wwwname{Sito Web}%
\providecaptionname{italian}\bankname{Conto bancario}%
\g@addnumerical@date{italian}{\number\day.\,\number\month.\,\number\year}%
\providecaptionname{spanish}\yourrefname{Su ref.}%
\providecaptionname{spanish}\yourmailname{Su carta de}%
\providecaptionname{spanish}\myrefname{Nuestra ref.}%
\providecaptionname{spanish}\customername{No. de cliente}%
\providecaptionname{spanish}\invoicename{No. de factura}%
\providecaptionname{spanish}\subjectname{Asunto}%
\providecaptionname{spanish}\ccname{Copias}%
\providecaptionname{spanish}\enclname{Adjunto}%
\providecaptionname{spanish}\headtoname{A}%
\providecaptionname{spanish}\headfromname{De}%
\providecaptionname{spanish}\datename{Fecha}%
\providecaptionname{spanish}\pagename{P\'agina}%
\providecaptionname{spanish}\phonename{Tel\'efono}%
\providecaptionname{spanish}\faxname{Fax}%
\providecaptionname{spanish}\emailname{Email}% or Correo electr\'onico
\providecaptionname{spanish}\wwwname{URL}% or P\`agina web
\providecaptionname{spanish}\bankname{Cuenta bancaria}%
\g@addnumerical@date{spanish}{\number\day.\,\number\month.\,\number\year}%
\providecaptionname{croatian}\yourrefname{Va\v{s} znak}%
\providecaptionname{croatian}\yourmailname{Va\v{s}e pismo od}%
\providecaptionname{croatian}\myrefname{Na\v{s} znak}%
\providecaptionname{croatian}\customername{Broj kupca}%
\providecaptionname{croatian}\invoicename{Broj fakture}%
\providecaptionname{croatian}\subjectname{Predmet}%
\providecaptionname{croatian}\ccname{Kopija}%
\providecaptionname{croatian}\enclname{Privitak}%
\providecaptionname{croatian}\headtoname{Prima}%
\providecaptionname{croatian}\headfromname{\v{S}alje}%
\providecaptionname{croatian}\datename{Nadnevak}%
\providecaptionname{croatian}\pagename{Stranica}%
\providecaptionname{croatian}\phonename{Telefon}%
\providecaptionname{croatian}\faxname{Fax}%
\providecaptionname{croatian}\emailname{E-Mail}%
\providecaptionname{croatian}\wwwname{URL}%
```

```
\providecaptionname{croatian}\bankname{Bankovna veza}%
\g@addnumerical@date{croatian}{\number\day.\,\number\month.\,\number\year.}%
\providecaptionname{dutch}\yourrefname{Uw kenmerk}%
\providecaptionname{dutch}\yourmailname{Uw brief van}%
\providecaptionname{dutch}\myrefname{Ons kenmerk}%
\providecaptionname{dutch}\customername{Klant No.}%
\providecaptionname{dutch}\invoicename{Rekening No.}%
\providecaptionname{dutch}\subjectname{Onderwerp}%
\providecaptionname{dutch}\ccname{Kopie aan}%
\providecaptionname{dutch}\enclname{Bijlage(n)}%
\providecaptionname{dutch}\headtoname{Aan}%
\providecaptionname{dutch}\headfromname{Van}%
\providecaptionname{dutch}\datename{Datum}%
\providecaptionname{dutch}\pagename{Pagina}%
\providecaptionname{dutch}\phonename{Telefoon}%
\providecaptionname{dutch}\faxname{Fax}%
\providecaptionname{dutch}\emailname{E--mail}%
\providecaptionname{dutch}\wwwname{URL}%
\providecaptionname{dutch}\bankname{Bankrekening}%
\g@addnumerical@date{dutch}{\number\day.\,\number\month.\,\number\year}%
\providecaptionname{finnish}\yourrefname{Viitteenne}%
\providecaptionname{finnish}\yourmailname{Kirjeenne}%
\providecaptionname{finnish}\myrefname{Viitteemme}%
\providecaptionname{finnish}\customername{Asiakasnumero}%
\providecaptionname{finnish}\invoicename{Laskun numero}%
\providecaptionname{finnish}\subjectname{Asia}%
\providecaptionname{finnish}\ccname{Jakelu}%
\providecaptionname{finnish}\enclname{Liitteet}%
\providecaptionname{finnish}\headtoname{Vastaanottaja}%
\providecaptionname{finnish}\headfromname{L\"ahett\"aj\"a}%
\providecaptionname{finnish}\datename{P\"aiv\"a}%
\providecaptionname{finnish}\pagename{Sivu}%
\providecaptionname{finnish}\phonename{Puhelin}%
\providecaptionname{finnish}\faxname{Faksi}%
\providecaptionname{finnish}\emailname{S\"ahk\"oposti}%
\providecaptionname{finnish}\wwwname{URL}%
\providecaptionname{finnish}\bankname{Pankkitilin numero}%
\g@addnumerical@date{finnish}{\number\day.\number\month.\number\year}%
\providecaptionname{norsk}\yourrefname{Deres ref.}%
\providecaptionname{norsk}\yourmailname{Deres brev av:}%
\providecaptionname{norsk}\myrefname{V\aa{}r ref:}%
\providecaptionname{norsk}\customername{Kundenummer}%
\providecaptionname{norsk}\invoicename{Fakturanummer}%
\providecaptionname{norsk}\subjectname{Emne}%
\providecaptionname{norsk}\ccname{Kopi til}%
\providecaptionname{norsk}\enclname{Vedlegg}%
\providecaptionname{norsk}\headtoname{Til}%
\providecaptionname{norsk}\headfromname{Fra}%
\providecaptionname{norsk}\datename{Dato}%
```

```
\providecaptionname{norsk}\pagename{Side}%
\providecaptionname{norsk}\phonename{Telefon}%
\providecaptionname{norsk}\faxname{Telefaks}%
\providecaptionname{norsk}\emailname{E-post}%
\providecaptionname{norsk}\wwwname{Url}%
\providecaptionname{norsk}\bankname{Bankkontonummer}%
\g@addnumerical@date{norsk}{\number\day.\number\month.\number\year}%
\providecaptionname{swedish}\yourrefname{Er ref}%
\providecaptionname{swedish}\yourmailname{Ert brev av}%
\providecaptionname{swedish}\myrefname{V\aa{}r ref}%
\providecaptionname{swedish}\customername{Kundnummer}%
\providecaptionname{swedish}\invoicename{Fakturanummer}%
\providecaptionname{swedish}\subjectname{\"Amne}%
```

„Kopia“ oder „Kopia till“ geht beides – da im Deutschen die lange Form verwendet wird nehmen wir die auch hier:

```
\providecaptionname{swedish}\ccname{Kopia till}%
```

„Bilaga“ ist Singular. Bei mehreren Anlagen schreibt man „Bilagor“. Da wir nicht wissen, ob es eine oder mehrere Anlagen sind, haben wir ein kleines Problem und gehen mal von mehreren aus:

```
\providecaptionname{swedish}\enclname{Bilagor}% Singular: Bilaga
\providecaptionname{swedish}\headtoname{Till}%
\providecaptionname{swedish}\headfromname{Fr\aa{}n}%
\providecaptionname{swedish}\datename{Datum}%
\providecaptionname{swedish}\pagename{Sida}%
\providecaptionname{swedish}\phonename{Telefon}%
\providecaptionname{swedish}\faxname{Telefax}%
\providecaptionname{swedish}\emailname{E-post}%
\providecaptionname{swedish}\wwwname{Hemsida}%
```

Für `\bankname` gibt es zwei verschiedene Ausdrücke, je nachdem um welche Kontoform es sich handelt: „Bankgiro“ und „PlusGiro“. Beide sind etwa gleich verbreitet, evtl. mit einem leichten Vorteil für Bankgiro.

```
\providecaptionname{swedish}\bankname{Bankgiro}% PlusGiro
```

Statt der ISO-Variante wird hier die traditionelle Variante, „18/1 2011“, verwendet.

```
\g@addnumerical@date{swedish}{\number\day/\number\month~\number\year}%
```

Zum Schluss findet noch die eigentliche Auswahl statt. Diese orientiert sich nun an der Auswahl nach german.sty 2.5b und verwendet keine festen Sprachzuordnungen mehr. Dafür sind nun keine Erweiterungen für andere Sprachen mehr möglich.

```
\captionsenglish
\dateenglish
```

Wird die `hyphen.cfg` von Babel verwendet, wird aber das *babel*-Paket nicht geladen, so ist `\languagename` häufig nicht korrekt, so dass dann

\selectlanguage mit \languagename als Argument schief geht. Daher hier ein Workaround für das Problem.

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname date\languagename\endcsname\relax
  \ClassWarningNoLine{%
    scrlttr2%
  }{%
    \string\language\space is \the\language, \string\languagename\space is
    `\languagename'\MessageBreak
    but \expandafter\string\csname
    date\languagename\endcsname\space not defined!\MessageBreak
    This seems to be a bug at you're `hyphen.cfg'.\MessageBreak
    Undefining macro \string\languagename\space to avoid errors%
  }%
  \let\languagename=\undefined
\fi
\ifx\languagename\undefined
  \ClassWarningNoLine{scrlttr2}{\string\languagename\space not
    defined, using \string\language.\MessageBreak
    This may result in use of wrong language!\MessageBreak
    You should use a compatible language
    package\MessageBreak
    (e.g. `Babel', `german', `ngerman', ...)}%
  \ifx\l@american\undefined\else\ifnum\language=\l@american
      \captionsamerican
      \dateamerican
  \fi\fi
  \ifx\l@british\undefined\else\ifnum\language=\l@british
      \captionsbritish
      \datebritish
  \fi\fi
  \ifx\l@UKenglish\undefined\else\ifnum\language=\l@UKenglish
      \captionsUKenglish
      \dateUKenglish
  \fi\fi
  \ifx\l@USenglish\undefined\else\ifnum\language=\l@USenglish
      \captionsUSenglish
      \dateUSenglish
  \fi\fi
  \ifx\l@austrian\undefined\else\ifnum\language=\l@austrian
      \captionsaustrian
      \dateaustrian
  \fi\fi
  \ifx\l@german\undefined\else\ifnum\language=\l@german
      \captionsgerman
      \dategerman
  \fi\fi
  \ifx\l@ngerman\undefined\else\ifnum\language=\l@ngerman
      \captionsngerman
```

```
        \datengerman
    \fi\fi
    \ifx\l@french\undefined\else\ifnum\language=\l@french
        \captionsfrench
        \datefrench
    \fi\fi
    \ifx\l@italian\undefined\else\ifnum\language=\l@italian
        \captionsitalian
        \dateitalian
    \fi\fi
    \ifx\l@spanish\undefined\else\ifnum\language=\l@spanish
        \captionsspanish
        \datespanish
    \fi\fi
    \ifx\l@croatian\undefined\else\ifnum\language=\l@croatian
        \captionscroatian
        \datecroatian
    \fi\fi
    \ifx\l@dutch\undefined\else\ifnum\language=\l@dutch
        \captionsdutch
        \datedutch
    \fi\fi
    \ifx\l@finnish\undefined\else\ifnum\language=\l@finnish
        \captionsfinnish
        \datefinnish
    \fi\fi
    \ifx\l@norsk\undefined\else\ifnum\language=\l@norsk
        \captionsnorsk
        \datenorsk
    \fi\fi
    \ifx\l@swedish\undefined\else\ifnum\language=\l@swedish
        \captionsswedish
        \dateswedish
    \fi\fi
  \else
    \edef\@tempa{nohyphenation}%
    \ifx\languagename\@tempa
      \ClassWarningNoLine{%
        scrlttr2%
      }{%
        You've selected language `\languagename'.\MessageBreak
        Maybe your LaTeX format contains Babel extension\MessageBreak
        but you have not selected a language using\MessageBreak
        Babel package.\MessageBreak
        Please select another language!\MessageBreak
        Only as a workaround english captions and date\MessageBreak
        will be used%
      }%
    \else
```

```
      \expandafter\selectlanguage\expandafter{\languagename}%
      \ClassInfo{%
        scrlttr2%
      }{%
        used language is ‘\languagename’.\MessageBreak
        Supported languages are ‘english’, ‘UKenglish’,\MessageBreak
        ‘british’, ‘american’, ‘USenglish’,\MessageBreak
        ‘german’, ‘ngerman’, ‘austrian’,\MessageBreak
        ‘dutch’, ‘italian’, ‘spanish’,\MessageBreak
        ‘croatian’, ‘finnish’, ‘norsk’%
      }%
    \fi
  \fi
}
```

Datei e
scrkfont.dtx

5 Schriftauswahl

Dieser Bereich befasst sich mit allem, was zur Schriftauswahl gehört. Die einzelnen Elemente, für die eine Schriftauswahl möglich ist, sind jedoch in den Dateien definiert, in denen diese Elemente auftreten.

5.1 Option zur Auswahl der Größe der Grundschrift

10pt
11pt
12pt

Diese Optionen existieren nicht mehr als explizite Optionen, sondern werden in `scrtarea.dtx` über `\DeclareOption*` ausgewertet.

\@ptsize

Wir definieren `\@ptsize` als Grundschriftgrößee minus 10 pt. Hier wird die Standardeinstellung 11 pt für alle Klassen außer der Briefklasse gewählt. Bis zum Einstellen tatsächlichen Einstellen der Schriftgröße wird in `\@ptsize` allerdings die tatsächliche Größe angegeben, damit sich Rundungsfehler nicht so leicht summieren.

```
⟨*!extend⟩
\newcommand*\@ptsize{%
⟨!letter⟩   11%
⟨letter⟩   12%
}
⟨/!extend⟩
```

\@pt@scan
\@pt@@scan

Das Macro `\@pt@scan` wird benötigt, um die Schriftgröße nach der alten Methode zu erkennen.

```
⟨*!extend⟩
\newcommand*{\@pt@scan}{%
  \expandafter\@pt@@scan\CurrentOption pt\@pt@@scan%
}
\newcommand*{\@pt@@scan}{}
\def\@pt@@scan #1pt#2\@pt@@scan{%
  \ifstr{#2}{pt}{%
    \ifnumber{#1}{%
      \KOMA@UseObsoleteOption{\PackageInfoNoLine{\KOMAClassName}}%
                                {#1#2}{fontsize=#1#2}%
    }{\@headlines}%
  }{\@headlines}%
}
⟨/!extend⟩
```

\@fontsizefilebase

Dieses Makro speichert den Präfix der primären Schriftgrößenoptionsdateien. Es wird nur definiert, wenn es nicht bereits definiert ist. Damit können

Wrapperklassen komplett andere Größen vorgeben, ohne mit `\ReplaceInput` arbeiten zu müssen.

```
\providecommand*{\@fontsizefilebase}{scrsize}
```

`fontsize` Hier nun die Option, die tatsächlich verwendet wird. Es sei darauf hingewiesen, dass damit auch Schriftgrößen wie `10.3pt` denkbar wären.

```
\KOMA@key{fontsize}{%
⟨*!extend⟩
  \scr@ifundefinedorrelax{changefontsizes}{%
    \@defaultunits\@tempdima#1pt\relax\@nnil
    \edef\@ptsize{#1}%
  }{%
⟨/!extend⟩
    \expandafter\@defaultunits\expandafter\@tempdima#1 pt\relax\@nnil
    \edef\@tempa{#1}%
    \setlength{\@tempdimb}{\@tempdima}%
    \edef\@tempb{\strip@pt\@tempdimb}%
    \addtolength{\@tempdimb}{-10\p@}%
    \edef\@ptsize{\strip@pt\@tempdimb}%
    \edef\@tempa{%
      \noexpand\makeatletter
      \noexpand\InputIfFileExists{\@fontsizefilebase\@tempa.clo}{%
⟨!extend⟩        \noexpand\ClassInfo{\KOMAClassName}{%
⟨extend⟩        \noexpand\PackageInfo{scrextend}{%
          File ‘\@fontsizefilebase\@tempa.clo’ used to setup font sizes}%
      }{%
        \noexpand\InputIfFileExists{\@fontsizefilebase\@tempb pt.clo}{%
⟨!extend⟩          \noexpand\ClassInfo{\KOMAClassName}{%
⟨extend⟩          \noexpand\PackageInfo{scrextend}{%
            File ‘\@fontsizefilebase\@tempb pt.clo’ used instead of%
            \noexpand\MessageBreak
            file ‘\@fontsizefilebase\@tempa.clo’ to setup font sizes}%
        }{%
          \noexpand\changefontsizes{#1}%
        }%
      }%
      \noexpand\catcode‘\noexpand\@=\the\catcode‘\@
    }%
⟨extend⟩  \scr@ifundefinedorrelax{changefontsizes}{%
⟨extend⟩    \expandafter\AtEndOfPackage\expandafter{\@tempa}%
⟨extend⟩  }{%
    \@tempa
  }%
}
```

5.2 Einlesen der Schriftgrößendatei

Trotzdem werden min. drei Schriftgrößendateien benötigt. Ab Version 2.98c verwendet KOMA-Script vorzugsweise eigene Dateien, die auch noch zu

einem späteren Zeitpunkt geladen werden können. Ansonsten sind diese Dateien weitestgehend dem Quellcode der Standardklassen `classes.dtx` entnommen.

```
\ProvidesFile{%
⟨10pt⟩  scrsize10pt.clo%
⟨11pt⟩  scrsize11pt.clo%
⟨12pt⟩  scrsize12pt.clo%
}[\KOMAScriptVersion\space font size class option %
⟨10pt⟩  (10pt)%
⟨11pt⟩  (11pt)%
⟨12pt⟩  (12pt)%
]
```

`\normalsize` `\small` `\footnotesize` `\scriptsize` `\tiny` `\large` `\Large` `\LARGE` `\huge` `\Huge`

Siehe `classes.dtx`.

```
\def\normalsize{%
⟨*10pt⟩
  \@setfontsize\normalsize\@xpt\@xiipt
  \abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
  \abovedisplayshortskip \z@ \@plus3\p@
  \belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
⟨/10pt⟩
⟨*11pt⟩
  \@setfontsize\normalsize\@xipt{13.6}%
  \abovedisplayskip 11\p@ \@plus3\p@ \@minus6\p@
  \abovedisplayshortskip \z@ \@plus3\p@
  \belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
⟨/11pt⟩
⟨*12pt⟩
  \@setfontsize\normalsize\@xiipt{14.5}%
  \abovedisplayskip 12\p@ \@plus3\p@ \@minus7\p@
  \abovedisplayshortskip \z@ \@plus3\p@
  \belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
⟨/12pt⟩
  \belowdisplayskip \abovedisplayskip
  \let\@listi\@listI
}
\def\small{%
⟨*10pt⟩
  \@setfontsize\small\@ixpt{11}%
  \abovedisplayskip 8.5\p@ \@plus3\p@ \@minus4\p@
  \abovedisplayshortskip \z@ \@plus2\p@
  \belowdisplayshortskip 4\p@ \@plus2\p@ \@minus2\p@
  \def\@listi{\leftmargin\leftmargini
    \topsep 4\p@ \@plus2\p@ \@minus2\p@
    \parsep 2\p@ \@plus\p@ \@minus\p@
    \itemsep \parsep}%
⟨/10pt⟩
⟨*11pt⟩
  \@setfontsize\small\@xpt\@xiipt
```

```
  \abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
  \abovedisplayshortskip \z@ \@plus3\p@
  \belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
  \def\@listi{\leftmargin\leftmargini
    \topsep 6\p@ \@plus2\p@ \@minus2\p@
    \parsep 3\p@ \@plus2\p@ \@minus\p@
    \itemsep \parsep}%
⟨/11pt⟩
⟨*12pt⟩
  \@setfontsize\small\@xipt{13.6}%
  \abovedisplayskip 11\p@ \@plus3\p@ \@minus6\p@
  \abovedisplayshortskip \z@ \@plus3\p@
  \belowdisplayshortskip 6.5\p@ \@plus3.5\p@ \@minus3\p@
  \def\@listi{\leftmargin\leftmargini
    \topsep 9\p@ \@plus3\p@ \@minus5\p@
    \parsep 4.5\p@ \@plus2\p@ \@minus\p@
    \itemsep \parsep}%
⟨/12pt⟩
  \belowdisplayskip \abovedisplayskip
}
\def\footnotesize{%
⟨*10pt⟩
  \@setfontsize\footnotesize\@viiipt{9.5}%
  \abovedisplayskip 6\p@ \@plus2\p@ \@minus4\p@
  \abovedisplayshortskip \z@ \@plus\p@
  \belowdisplayshortskip 3\p@ \@plus\p@ \@minus2\p@
  \def\@listi{\leftmargin\leftmargini
    \topsep 3\p@ \@plus\p@ \@minus\p@
    \parsep 2\p@ \@plus\p@ \@minus\p@
    \itemsep \parsep}%
⟨/10pt⟩
⟨*11pt⟩
  \@setfontsize\footnotesize\@ixpt{11}%
  \abovedisplayskip 8\p@ \@plus2\p@ \@minus4\p@
  \abovedisplayshortskip \z@ \@plus\p@
  \belowdisplayshortskip 4\p@ \@plus2\p@ \@minus2\p@
  \def\@listi{\leftmargin\leftmargini
    \topsep 4\p@ \@plus2\p@ \@minus2\p@
    \parsep 2\p@ \@plus\p@ \@minus\p@
    \itemsep \parsep}%
⟨/11pt⟩
⟨*12pt⟩
   \@setfontsize\footnotesize\@xpt\@xiipt
   \abovedisplayskip 10\p@ \@plus2\p@ \@minus5\p@
   \abovedisplayshortskip \z@ \@plus3\p@
   \belowdisplayshortskip 6\p@ \@plus3\p@ \@minus3\p@
   \def\@listi{\leftmargin\leftmargini
               \topsep 6\p@ \@plus2\p@ \@minus2\p@
               \parsep 3\p@ \@plus2\p@ \@minus\p@
```

```
                \itemsep \parsep}%
⟨/12pt⟩
  \belowdisplayskip \abovedisplayskip
}
⟨*10pt⟩
\def\scriptsize{\@setfontsize\scriptsize\@viipt\@viiipt}
\def\tiny{\@setfontsize\tiny\@vpt\@vipt}
\def\large{\@setfontsize\large\@xiipt{14}}
\def\Large{\@setfontsize\Large\@xivpt{18}}
\def\LARGE{\@setfontsize\LARGE\@xviipt{22}}
\def\huge{\@setfontsize\huge\@xxpt{25}}
\def\Huge{\@setfontsize\Huge\@xxvpt{30}}
⟨/10pt⟩
⟨*11pt⟩
\def\scriptsize{\@setfontsize\scriptsize\@viiipt{9.5}}
\def\tiny{\@setfontsize\tiny\@vipt\@viipt}
\def\large{\@setfontsize\large\@xiipt{14}}
\def\Large{\@setfontsize\Large\@xivpt{18}}
\def\LARGE{\@setfontsize\LARGE\@xviipt{22}}
\def\huge{\@setfontsize\huge\@xxpt{25}}
\def\Huge{\@setfontsize\Huge\@xxvpt{30}}
⟨/11pt⟩
⟨*12pt⟩
\def\scriptsize{\@setfontsize\scriptsize\@viiipt{9.5}}
\def\tiny{\@setfontsize\tiny\@vipt\@viipt}
\def\large{\@setfontsize\large\@xivpt{18}}
\def\Large{\@setfontsize\Large\@xviipt{22}}
\def\LARGE{\@setfontsize\LARGE\@xxpt{25}}
\def\huge{\@setfontsize\huge\@xxvpt{30}}
\let\Huge=\huge
⟨/12pt⟩
\normalsize
```

\footnotesep
\footins
\floatsep
\textfloatsep
\intextsep
\dblfloatsep
\sbltextfloatsep
\@fptop
\@fpsep
\@fpbot
\@dblfptop
\@dblfpsep
\@dblfpbot
\partopsep

Siehe `classes.dtx`.

```
⟨*10pt⟩
\setlength\footnotesep    {6.65\p@}
\setlength{\skip\footins} {9\p@ \@plus 4\p@ \@minus 2\p@}
\setlength\floatsep       {12\p@ \@plus 2\p@ \@minus 2\p@}
\setlength\textfloatsep   {20\p@ \@plus 2\p@ \@minus 4\p@}
\setlength\intextsep      {12\p@ \@plus 2\p@ \@minus 2\p@}
\setlength\dblfloatsep    {12\p@ \@plus 2\p@ \@minus 2\p@}
\setlength\dbltextfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
\setlength\@fptop         {0\p@ \@plus 1fil}
\setlength\@fpsep         {8\p@ \@plus 2fil}
\setlength\@fpbot         {0\p@ \@plus 1fil}
\setlength\@dblfptop      {0\p@ \@plus 1fil}
\setlength\@dblfpsep      {8\p@ \@plus 2fil}
\setlength\@dblfpbot      {0\p@ \@plus 1fil}
\setlength\partopsep      {2\p@ \@plus 1\p@ \@minus 1\p@}
```

```
⟨/10pt⟩
⟨*11pt⟩
\setlength\footnotesep    {7.7\p@}
\setlength{\skip\footins} {10\p@ \@plus 4\p@ \@minus 2\p@}
\setlength\floatsep       {12\p@ \@plus 2\p@ \@minus 2\p@}
\setlength\textfloatsep   {20\p@ \@plus 2\p@ \@minus 4\p@}
\setlength\intextsep      {12\p@ \@plus 2\p@ \@minus 2\p@}
\setlength\dblfloatsep    {12\p@ \@plus 2\p@ \@minus 2\p@}
\setlength\dbltextfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
\setlength\@fptop         {0\p@ \@plus 1fil}
\setlength\@fpsep         {8\p@ \@plus 2fil}
\setlength\@fpbot         {0\p@ \@plus 1fil}
\setlength\@dblfptop      {0\p@ \@plus 1fil}
\setlength\@dblfpsep      {8\p@ \@plus 2fil}
\setlength\@dblfpbot      {0\p@ \@plus 1fil}
\setlength\partopsep      {3\p@ \@plus 1\p@ \@minus 1\p@}
⟨/11pt⟩
⟨*12pt⟩
\setlength\footnotesep    {8.4\p@}
\setlength{\skip\footins} {10.8\p@ \@plus 4\p@ \@minus 2\p@}
\setlength\floatsep       {12\p@ \@plus 2\p@ \@minus 4\p@}
\setlength\textfloatsep   {20\p@ \@plus 2\p@ \@minus 4\p@}
\setlength\intextsep      {14\p@ \@plus 4\p@ \@minus 4\p@}
\setlength\dblfloatsep    {14\p@ \@plus 2\p@ \@minus 4\p@}
\setlength\dbltextfloatsep{20\p@ \@plus 2\p@ \@minus 4\p@}
\setlength\@fptop         {0\p@ \@plus 1fil}
\setlength\@fpsep         {10\p@ \@plus 2fil}
\setlength\@fpbot         {0\p@ \@plus 1fil}
\setlength\@dblfptop      {0\p@ \@plus 1fil}
\setlength\@dblfpsep      {10\p@ \@plus 2fil}
\setlength\@dblfpbot      {0\p@ \@plus 1fil}
\setlength\partopsep      {3\p@ \@plus 2\p@ \@minus 2\p@}
⟨/12pt⟩
```

`\@listi`
`\@listii`
`\@listiii`
`\@listiv`
`\@listv`
`\@listvi`

Siehe `classes.dtx`.

```
\def\@listi{\leftmargin\leftmargini
⟨*10pt⟩
            \parsep 4\p@ \@plus2\p@ \@minus\p@
            \topsep 8\p@ \@plus2\p@ \@minus4\p@
            \itemsep4\p@ \@plus2\p@ \@minus\p@}
⟨/10pt⟩
⟨*11pt⟩
            \parsep 4.5\p@ \@plus2\p@ \@minus\p@
            \topsep 9\p@   \@plus3\p@ \@minus5\p@
            \itemsep4.5\p@ \@plus2\p@ \@minus\p@}
⟨/11pt⟩
⟨*12pt⟩
            \parsep 5\p@  \@plus2.5\p@ \@minus\p@
            \topsep 10\p@ \@plus4\p@   \@minus6\p@
```

```
            \itemsep5\p@  \@plus2.5\p@ \@minus\p@}
⟨/12pt⟩
\let\@listI\@listi
\def\@listii {\leftmargin\leftmarginii
              \labelwidth\leftmarginii
              \advance\labelwidth-\labelsep
⟨*10pt⟩
              \topsep    4\p@ \@plus2\p@ \@minus\p@
              \parsep    2\p@ \@plus\p@  \@minus\p@
⟨/10pt⟩
⟨*11pt⟩
              \topsep    4.5\p@ \@plus2\p@ \@minus\p@
              \parsep    2\p@   \@plus\p@  \@minus\p@
⟨/11pt⟩
⟨*12pt⟩
              \topsep    5\p@   \@plus2.5\p@ \@minus\p@
              \parsep    2.5\p@ \@plus\p@    \@minus\p@
⟨/12pt⟩
              \itemsep   \parsep}
\def\@listiii{\leftmargin\leftmarginiii
              \labelwidth\leftmarginiii
              \advance\labelwidth-\labelsep
⟨10pt⟩              \topsep    2\p@ \@plus\p@\@minus\p@
⟨11pt⟩              \topsep    2\p@ \@plus\p@\@minus\p@
⟨12pt⟩              \topsep    2.5\p@\@plus\p@\@minus\p@
              \parsep    \z@
              \partopsep \p@ \@plus\z@ \@minus\p@
              \itemsep   \topsep}
\def\@listiv {\leftmargin\leftmarginiv
              \labelwidth\leftmarginiv
              \advance\labelwidth-\labelsep}
\def\@listv  {\leftmargin\leftmarginv
              \labelwidth\leftmarginv
              \advance\labelwidth-\labelsep}
\def\@listvi {\leftmargin\leftmarginvi
              \labelwidth\leftmarginvi
              \advance\labelwidth-\labelsep}
```

`\changefontsizes` Diese Anweisung bietet berechnete Schriftgrößen als Fallbacklösung an. Das optionale erste Argument ist dabei der Grundlinienabstand. Das zweite Argument ist die gewünschte Grundschriftgröße.

```
\newcommand*{\changefontsizes}[2][1.2\@tempdima]{%
⟨!extend⟩  \ClassWarning{\KOMAClassName}{%
⟨extend⟩  \PackageWarning{scrextend}{%
    Using fallback calculation to setup font sizes\MessageBreak
    for basic size ‘#2’}%
  \scr@ifundefinedorrelax{dimexpr}{%
⟨!extend⟩    \ClassError{\KOMAClassName}{%
⟨extend⟩    \PackageError{scrextend}{%
```

```
      feature only allowed using e-TeX%
    }{%
      The fallback calculation to setup font sizes was implemented
      using e-TeX.\MessageBreak
      You should not continue, because the fallback calculation would fail%
    }
  }{%
    \@defaultunits\@tempdima#2pt\relax\@nnil
    \setlength{\@tempdimc}{\@tempdima}%
    \addtolength{\@tempdimc}{-10\p@}%
    \edef\@ptsize{\strip@pt\@tempdimc}%
    \@defaultunits\@tempdimb#1pt\relax\@nnil
    \setlength{\@tempdimc}{\dimexpr (100\@tempdimb / \@tempdima * \p@)}%
    \edef\@tempb{\the\@tempdimc}%
    \setlength{\@tempdimc}{\@tempdima}%
    \def\@tempa##1##2##3##4##5\@nnil{\def##1{##2.##3##4}}%
    \expandafter\@tempa\expandafter\@tempb\@tempb\@nnil%
```

Ab hier ist `\@tempdimc` die gewünschte Grundschriftgröße in pt und `\@tempb` der Faktor für den Grundlinienabstand auf 2 Stellen hinter dem Komma.

```
    \setlength{\abovedisplayskip}{%
      \@tempdima \@plus .25\@tempdima \@minus .58\@tempdima}%
    \setlength{\abovedisplayshortskip}{\z@ \@plus .25\@tempdima}%
    \setlength{\belowdisplayskip}{\abovedisplayskip}%
    \setlength{\belowdisplayshortskip}{%
      .55\@tempdima \@plus .3\@tempdima \@minus .25\@tempdima}%
    \edef\normalsize{%
      \noexpand\@setfontsize\noexpand\normalsize
      {\the\@tempdima}{\the\@tempdimb}%
      \abovedisplayskip \the\abovedisplayskip
      \abovedisplayshortskip \the\abovedisplayshortskip
      \belowdisplayskip \the\belowdisplayskip
      \belowdisplayshortskip \the\belowdisplayshortskip
      \let\noexpand\@listi\noexpand\@listI
    }%
    \setlength{\@tempdima}{0.9125\@tempdimc}%
    \setlength{\@tempdimb}{\@tempb\@tempdima}%
    \setlength{\abovedisplayskip}{%
      \@tempdima \@plus .25\@tempdima \@minus .58\@tempdima}%
    \setlength{\abovedisplayshortskip}{\z@ \@plus .25\@tempdima}%
    \setlength{\belowdisplayskip}{\abovedisplayskip}%
    \setlength{\belowdisplayshortskip}{%
      .55\@tempdima \@plus .3\@tempdima \@minus .25\@tempdima}%
    \edef\small{%
      \noexpand\@setfontsize\noexpand\small
      {\the\@tempdima}{\the\@tempdimb}%
      \abovedisplayskip \the\abovedisplayskip
      \abovedisplayshortskip \the\abovedisplayshortskip
      \belowdisplayskip \the\belowdisplayskip
```

```
    \belowdisplayshortskip \the\belowdisplayshortskip
    \let\noexpand\@listi\noexpand\@listi@small
  }%
  \setlength{\@tempdima}{.83334\@tempdimc}%
  \setlength{\@tempdimb}{\@tempb\@tempdima}%
  \setlength{\abovedisplayskip}{%
    \@tempdima \@plus .25\@tempdima \@minus .58\@tempdima}%
  \setlength{\abovedisplayshortskip}{\z@ \@plus .25\@tempdima}%
  \setlength{\belowdisplayskip}{\abovedisplayskip}%
  \setlength{\belowdisplayshortskip}{%
    .55\@tempdima \@plus .3\@tempdima \@minus .25\@tempdima}%
  \edef\footnotesize{%
    \noexpand\@setfontsize\noexpand\footnotesize
    {\the\@tempdima}{\the\@tempdimb}%
    \abovedisplayskip \the\abovedisplayskip
    \abovedisplayshortskip \the\abovedisplayshortskip
    \belowdisplayskip \the\belowdisplayskip
    \belowdisplayshortskip \the\belowdisplayshortskip
    \let\noexpand\@listi\noexpand\@listi@footnotesize
  }%
  \setlength{\@tempdima}{.66667\@tempdimc}%
  \setlength{\@tempdimb}{\@tempb\@tempdima}%
  \edef\scriptsize{%
    \noexpand\@setfontsize\noexpand\scriptsize
    {\the\@tempdima}{\the\@tempdimb}%
  }%
  \setlength{\@tempdima}{.5\@tempdimc}%
  \setlength{\@tempdimb}{\@tempb\@tempdima}%
  \edef\tiny{%
    \noexpand\@setfontsize\noexpand\tiny
    {\the\@tempdima}{\the\@tempdimb}%
  }%
  \setlength{\@tempdima}{1.2\@tempdimc}%
  \setlength{\@tempdimb}{\@tempb\@tempdima}%
  \edef\large{%
    \noexpand\@setfontsize\noexpand\large
    {\the\@tempdima}{\the\@tempdimb}%
  }%
  \setlength{\@tempdima}{1.44\@tempdimc}%
  \setlength{\@tempdimb}{\@tempb\@tempdima}%
  \edef\Large{%
    \noexpand\@setfontsize\noexpand\Large
    {\the\@tempdima}{\the\@tempdimb}%
  }%
  \setlength{\@tempdima}{1.728\@tempdimc}%
  \setlength{\@tempdimb}{\@tempb\@tempdima}%
  \edef\LARGE{%
    \noexpand\@setfontsize\noexpand\LARGE
    {\the\@tempdima}{\the\@tempdimb}%
```

```
  }%
  \setlength{\@tempdima}{2.074\@tempdimc}%
  \setlength{\@tempdimb}{\@tempb\@tempdima}%
  \edef\huge{%
    \noexpand\@setfontsize\noexpand\huge
    {\the\@tempdima}{\the\@tempdimb}%
  }%
  \setlength{\@tempdima}{2.488\@tempdimc}%
  \setlength{\@tempdimb}{\@tempb\@tempdima}%
  \edef\Huge{%
    \noexpand\@setfontsize\noexpand\Huge
    {\the\@tempdima}{\the\@tempdimb}%
  }%
  \normalsize
  \setlength{\footnotesep}{.7\@tempdimc}%
  \setlength{\skip\footins}{.9\@tempdimc \@plus .3333\@tempdimc \@minus
    .6\@tempdimc}%
  \setlength{\floatsep}{\@tempdimc \@plus .1667\@tempdimc \@minus
    .3333\@tempdimc}%
  \setlength{\textfloatsep}{1.6667\@tempdimc \@plus .1667\@tempdimc \@minus
    .3333\@tempdimc}%
  \setlength{\intextsep}{\@tempb\@tempdimc \@plus .3333\@tempdimc \@minus
    .3333\@tempdimc}%
  \setlength{\dblfloatsep}{\@tempb\@tempdimc \@plus .1667\@tempdimc \@minus
    .3333\@tempdimc}%
  \setlength{\dbltextfloatsep}{\textfloatsep}%
  \setlength{\@fptop}{0\p@ \@plus 1fil}%
  \setlength{\@fpsep}{.8333\@tempdimc \@plus 2fil}%
  \setlength{\@fpbot}{\@fptop}%
  \setlength{\@dblfptop}{0\p@ \@plus 1fil}%
  \setlength{\@dblfpsep}{.8333\@tempdimc \@plus 2fil}%
  \setlength{\@dblfpbot}{\@fptop}%
  \setlength{\partopsep}{.25\@tempdimc \@plus .16667\@tempdimc \@minus
    .16667\@tempdimc}%
  \setlength{\topsep}{.8333\@tempdimc \@plus .3333\@tempdimc \@minus
    .5\@tempdimc}%
  \setlength{\parsep}{.4167\@tempdimc \@plus .2083\@tempdimc \@minus \p@}%
  \@tempswafalse
  \begingroup
    \def\@list@extra{\aftergroup\@tempswatrue}%
    \csname @listi\endcsname
  \endgroup
  \edef\@listi{\leftmargin\leftmargini
    \topsep \the\topsep
    \parsep \the\parsep
    \itemsep \parsep
    \if@tempswa\noexpand\@list@extra\fi
  }%
  \let\@listI\@listi
```

```
  \setlength{\topsep}{.75\@tempdimc \@plus .25\@tempdimc \@minus
    .41667\@tempdimc}%
  \setlength{\parsep}{.375\@tempdimc \@plus .16667\@tempdimc \@minus \p@}%
  \@tempswafalse
  \begingroup
    \def\@list@extra{\aftergroup\@tempswatrue}%
    \csname @listi\endcsname
  \endgroup
  \edef\@listi@small{\leftmargin\leftmargini
    \topsep \the\topsep
    \parsep \the\parsep
    \itemsep \parsep
    \if@tempswa\noexpand\@list@extra\fi
  }%
  \setlength{\topsep}{.5\@tempdimc \@plus .16667\@tempdimc \@minus
    .16667\@tempdimc}%
  \setlength{\parsep}{.25\@tempdimc \@plus .16667\@tempdimc \@minus \p@}%
  \@tempswafalse
  \begingroup
    \def\@list@extra{\aftergroup\@tempswatrue}%
    \csname @listi\endcsname
  \endgroup
  \edef\@listi@footnotesize{\leftmargin\leftmargini
    \topsep \the\topsep
    \parsep \the\parsep
    \itemsep \parsep
    \if@tempswa\noexpand\@list@extra\fi
  }%
  \setlength{\topsep}{.4167\@tempdimc \@plus .2083\@tempdimc \@minus \p@}%
  \setlength{\parsep}{.2083\@tempdimc \@plus \p@ \@minus \p@}%
  \@tempswafalse
  \begingroup
    \def\@list@extra{\aftergroup\@tempswatrue}%
    \csname @listii\endcsname
  \endgroup
  \edef\@listii{\leftmargin\leftmarginii
    \labelwidth=\dimexpr \leftmargin-\labelsep
    \topsep \the\topsep
    \parsep \the\parsep
    \itemsep \parsep
    \if@tempswa\noexpand\@list@extra\fi
  }%
  \setlength{\topsep}{.2083\@tempdimc \@plus \p@ \@minus \p@}%
  \setlength{\parsep}{\p@ \@plus\z@ \@minus\p@}%
  \@tempswafalse
  \begingroup
    \def\@list@extra{\aftergroup\@tempswatrue}%
    \csname @listiii\endcsname
  \endgroup
```

```
    \edef\@listiii{\leftmargin\leftmarginiii
      \labelwidth=\dimexpr \leftmargin-\labelsep
      \topsep \the\topsep
      \parsep \z@
      \partopsep \the\parsep
      \itemsep \topsep
      \if@tempswa\noexpand\@list@extra\fi
    }%
    \@tempswafalse
    \begingroup
      \def\@list@extra{\aftergroup\@tempswatrue}%
      \csname @listiv\endcsname
    \endgroup
    \def\@listiv{\leftmargin\leftmarginiv
      \labelwidth=\dimexpr \leftmargin-\labelsep
      \if@tempswa\noexpand\@list@extra\fi
    }%
    \@tempswafalse
    \begingroup
      \def\@list@extra{\aftergroup\@tempswatrue}%
      \csname @listv\endcsname
    \endgroup
    \def\@listv{\leftmargin\leftmarginv
      \labelwidth=\dimexpr \leftmargin-\labelsep
      \if@tempswa\noexpand\@list@extra\fi
    }%
    \@tempswafalse
    \begingroup
      \def\@list@extra{\aftergroup\@tempswatrue}%
      \csname @listvi\endcsname
    \endgroup
    \def\@listvi{\leftmargin\leftmarginvi
      \labelwidth=\dimexpr \leftmargin-\labelsep
      \if@tempswa\noexpand\@list@extra\fi
    }%
    \@listi
```

Einige Platzierungsabstände sind ebenfalls schriftgrößenabhängig:

```
    \setlength{\@tempdimb}{\@tempb\@tempdima}%
    \setlength{\@tempdima}{\dimexpr \@tempdimb-\@tempdimc}%
    \setlength\intextsep{\@tempdimb \@plus.2\@tempdima \@minus.2\@tempdima}%
    \setlength\dblfloatsep\intextsep
    \setlength\partopsep{.2\@tempdimc \@plus.1\@tempdimc \@minus.1\@tempdimc}%
  }%
}
```

Zunächst wird die gewünschte Größe, die in `\@ptsize` abgelegt ist, als Länge ermittelt. In `\@tempa` wird die ursprünglich gewünschte Größe abgelegt, damit zwischen `10pt`, `10` und `10dd` unterschieden werden kann. In `\@tempb` wird hingegen die Größe in pt abgelegt, damit die Standard-

Dateien verwendet werden können. Dann wird `\@ptsize` schon einmal richtig eingestellt.

```
⟨*!extend⟩
\expandafter\@defaultunits\expandafter\@tempdima\@ptsize pt\relax\@nnil
\edef\@tempa{\@ptsize}%
\setlength{\@tempdimb}{\@tempdima}%
\edef\@tempb{\strip@pt\@tempdimb}%
\addtolength{\@tempdimb}{-10\p@}%
\edef\@ptsize{\strip@pt\@tempdimb}%
```

Jetzt wird zunächst versucht, ob eine Datei „`\fontsizefilebase\@tempa.clo`“ vorhanden ist, also eine, bei der die ursprüngliche Einheit im Namen angegeben ist. Wenn das der Fall ist, wird diese verwendet. Wenn nicht wird das Ganze für `\@tempb` um die Einheit pt erweitert wiederholt.

```
\InputIfFileExists{\@fontsizefilebase\@tempa.clo}{
  \ClassInfo{\KOMAClassName}{%
    File ‘\@fontsizefilebase\@tempa.clo’ used to setup font sizes}%
}{%
  \InputIfFileExists{\@fontsizefilebase\@tempb pt.clo}{%
    \ClassInfo{\KOMAClassName}{%
      File ‘\@fontsizefilebase\@tempb pt.clo’ used instead of\MessageBreak
      file ‘\@fontsizefilebase\@tempa.clo’ to setup font sizes}%
  }{%
```

Wurde bis hier noch keine Datei geladen, wird nun versucht, die Größendateien der Standardklassen zu verwenden.

```
⟨*book⟩
    \InputIfFileExists{bk\@tempb.clo}{%
      \ClassInfo{\KOMAClassName}{%
        File ‘bk\@tempb.clo’ used to setup font sizes}%
    }{%
⟨/book⟩
      \InputIfFileExists{size\@tempb.clo}{%
        \ClassInfo{\KOMAClassName}{%
          File ‘size\@tempb.clo’ used to setup font sizes}%
      }{%
```

Sind auch diese Dateien nicht vorhanden, so wird als Fallback eine Berechnung der Schriftgrößen verwendet. Es sei darauf hingewiesen, dass dabei auch `\@tempa`, `\@tempb` und `\@ptsize` neu gesetzt werden.

```
        \changefontsizes{\@tempa}%
```

Damit sollten nun die Schriftgrößen eingestellt sein.

```
      }%
⟨book⟩    }%
  }%
}
⟨/!extend⟩
```

5.3 „Alte“ Font-Auswahlbefehle

\rm \sf \tt \bf \it \sl \sc \sfb

Um die Umstellung von LaTeX auf LaTeX 2_ε zu erleichtern, gibt es die alten Font-Auswahlbefehle `\rm`, `\sf`, `\tt`, `\bf`, `\it`, `\sl` und `\sc` auch in den Klassen. Es ist zu beachten, dass diese Befehle hier nach dem alten Fontauswahlverfahren arbeiten. Es werden also immer alle Parameter zugleich geändert. Somit ist zu empfehlen, dass statt dieser Befehle zukünftig in der Regel die neuen `\text...`-Befehle verwendet werden.

Der aus der Script 2.0-Familie bekannte Befehl `\sfb` ist hier ebenfalls definiert.

```
⟨*!extend⟩
\DeclareOldFontCommand{\rm}{\normalfont\rmfamily}{\mathrm}
\DeclareOldFontCommand{\sf}{\normalfont\sffamily}{\mathsf}
\DeclareOldFontCommand{\tt}{\normalfont\ttfamily}{\mathtt}
\DeclareOldFontCommand{\bf}{\normalfont\bfseries}{\mathbf}
\DeclareOldFontCommand{\it}{\normalfont\itshape}{\mathit}
\DeclareOldFontCommand{\sl}{\normalfont\slshape}{\@nomath\sl}
\DeclareOldFontCommand{\sc}{\normalfont\scshape}{\@nomath\sc}
\DeclareOldFontCommand{\sfb}{\normalfont\sffamily\bfseries}{%
  \@nomath\sfb}
⟨/!extend⟩
```

\cal

Der Befehl ist in LaTeX nicht mehr definiert. Deshalb wird er nun neu und gleich robust deklariert. Dafür fällt `\pcal` weg.

\mit

Der Befehl ist in LaTeX nicht mehr definiert. Deshalb wird er nun neu und gleich robust deklariert. Dafür fällt `\pmit` weg.

Die beiden Befehle `\cal` und `\mit` war bis zur LaTeX-Version vom 1.12.1994 patch level 3 noch im Kernal definiert. Jetzt sind die beiden den Klassen überlassen. Hier ist die Definition aus den Standard-classes übernommen.

```
⟨*!extend⟩
\DeclareRobustCommand*{\cal}{\@fontswitch\relax\mathcal}
\DeclareRobustCommand*{\mit}{\@fontswitch\relax\mathnormal}
⟨/!extend⟩
```

5.4 Setzen der Schrift eines Elements

\setkomafont

Mit Hilfe dieses Makros kann die Schriftart von KOMA-Script-Elementen gesetzt werden.

```
\newcommand*{\setkomafont}[2]{%
  \@ifundefined{scr@fnt@#1}{%
    \@ifundefined{scr@fnt@instead@#1}{%
      \PackageError{scrkbase}{%
        font of element ‘#1’ can’t be set}{%
        You’ve told me to redefine the font selection of the
        element,\MessageBreak%
```

```
        but either no such element is known by
        KOMA-Script\MessageBreak%
        or the element does not use a special font selection%
      }%
    }{%
      \PackageInfo{scrkbase}{%
        You've told me to redefine the font selection of the\MessageBreak
        element `#1' that is an alias of element\MessageBreak
        `\csname scr@fnt@instead@#1\endcsname'%
      }%
      \expandafter\setkomafont\expandafter{%
        \csname scr@fnt@instead@#1\endcsname}{#2}%
    }%
  }{%
    \expandafter\expandafter\expandafter\def\csname scr@fnt@#1\endcsname{#2}%
  }%
  \@ifundefined{scr@fnt@wrn@#1}{}{%
    \PackageWarning{scrkbase}{%
      \csname scr@fnt@wrn@#1\endcsname{#1}%
    }%
  }%
}
```

`\addtokomafont` Dieses Makro funktioniert fast wie obiges, allerdings fügt es der vorhandenen Definition etwas an. Dazu wird `\l@addto@macro` aus scrkbase verwendet.

```
\newcommand*{\addtokomafont}[2]{%
  \@ifundefined{scr@fnt@#1}{%
    \@ifundefined{scr@fnt@instead@#1}{%
      \PackageError{scrkbase}{%
        font of element `#1' can't be extended%
      }{%
        You've told me to extend the font selection of the
        element,\MessageBreak
        but either no such element is known by
        KOMA-Script\MessageBreak
        or the element does not use a special font selection%
      }%
    }{%
      \PackageInfo{scrkbase}{%
        You've told me to extend the font selection of the\MessageBreak
        element `#1' that is an alias of element\MessageBreak
        `\csname scr@fnt@instead@#1\endcsname'%
      }%
      \expandafter\addtokomafont\expandafter{%
        \csname scr@fnt@instead@#1\endcsname}{#2}%
    }%
  }{%
    \expandafter\expandafter\expandafter\l@addto@macro
```

```
    \csname scr@fnt@#1\endcsname{#2}%
  }%
  \@ifundefined{scr@fnt@wrn@#1}{}{%
    \PackageWarning{scrkbase}{%
      \csname scr@fnt@wrn@#1\endcsname{#1}%
    }%
  }%
}
```

\usekomafont Mit Hilfe dieses Makros kann auf die Schriftart von KOMA-Script-Elementen umgeschaltet werden.

```
\newcommand*{\usekomafont}[1]{%
  \@ifundefined{scr@fnt@#1}{%
    \@ifundefined{scr@fnt@instead@#1}{%
      \PackageError{scrkbase}{%
        font of element `#1' can't be used%
      }{%
        You've told me to use the font selection of the
        element,\MessageBreak
        but either no such element is known by
        KOMA-Script\MessageBreak
        or the element does not use a special font selection%
      }%
    }{%
      \PackageInfo{scrkbase}{%
        You've told me to use the font selection of the
        element\MessageBreak
        `#1' that is an alias of element `\csname
        scr@fnt@instead@#1\endcsname'\MessageBreak
      }%
      \expandafter\expandafter\expandafter\usekomafont
      \expandafter\expandafter\expandafter{%
        \csname scr@fnt@instead@#1\endcsname}%
    }%
  }{%
    \@nameuse{scr@fnt@#1}%
  }%
}
```

Um dann ein Element zu definieren, definiert man zunächst ein Makro, das die Schriftart enthält. Dann definiert man `\scr@fnt@`*`Elementname`*. Wobei der Inhalt dieses Makros das Makro ist, das die Schriftart speichert. Damit ist das Element definiert. Soll ein Element durch ein anderes Element gesteuert werden, so definiert man stattdessen `\scr@fnt@instead@`*`Elementname`*, wobei der Inhalt des Makros dann der Name jenes anderen Elements ist. Soll beim Ändern der Schrift eines Elements eine zusätzliche Warnung ausgegeben werden, so kann der Text dieser Warnung in `\scr@fnt@wrn@`*`Elementname`* abgelegt werden. Also alles eigentlich ganz einfach. Damit es noch einfacher

wird, ein paar Hilfsmakros:

\newkomafont Mit \newkomafont wird ein neues Element definiert. Das erste optionale Argument ist dabei die optionale Warnung. Ist dieses nicht gesetzt oder \relax, so wird keine Warnung definiert. Das erste obligatorische Argument ist der Name des Elements. Als letztes folgt die Voreinstellung für die Schrift dieses Elements. Als Fontmakro wird übrigens \@*Elementname*font definiert.

```
\newcommand*{\newkomafont}[3][\relax]{%
  \expandafter\newcommand\expandafter*\csname @#2font\endcsname{#3}%
  \expandafter\expandafter\expandafter\newcommand
  \expandafter\expandafter\expandafter*%
  \expandafter\csname scr@fnt@#2\expandafter\endcsname\expandafter{%
    \csname @#2font\endcsname%
  }%
  \ifx\relax#1\relax\else
    \expandafter\newcommand\expandafter*\csname scr@fnt@wrn@#2\endcsname{#1}%
  \fi
}
```

\aliaskomafont Mit \aliaskomafont wird hingegen einfach ein Alias definiert. Dieser Befehlt kennt zwei Argumente: der Name des Elements und der Names des Elements, das stattdessen verwendet werden soll.

```
\newcommand*{\aliaskomafont}[2]{%
  \expandafter\newcommand\expandafter*\csname scr@fnt@instead@#1\endcsname{%
    #2%
  }%
}
```

Datei f scrktare.dtx

6 Satzspiegel Makros

Dieser Abschnitt beinhaltet alle Makros, die für den Satzspiegel benötigt werden. Dies betrifft sowohl das Paket typearea als auch die Klassen.

6.1 Präambel

Zuerst einmal zeigen, was das Paket ist:

```
\typeout{%
  Package typearea, \csname ver@typearea.sty\endcsname^^J%
  \@spaces\@spaces\@spaces\@spaces\space\space
  Copyright (C) Frank Neukam, 1992-1994^^J%
  \@spaces\@spaces\@spaces\@spaces\space\space
  Copyright (C) Markus Kohm, 1994-^^J%
}
```

6.2 Von den Optionen benötigte Befehle

Für das Paket

⟨∗package⟩

Zur Auswertung der Optionen werden einige Makros benötigt. Diese sind natürlich vor der Deklaration der Optionen bzw. vor der Auswertung der Optionen zu definieren. Ich mache das hier vor der Deklaration der Optionen.

`\SetDIVList` Der Befehl `\SetDIVList` tut nichts weiter als die innere Variable für die schriftgrößenabhängige *DIV*-Liste `\ta@divlist` neu zu definieren. Dabei können *DIV*-Werte für Größen ab 10 pt gesetzt werden.

```
\newcommand*\SetDIVList[1]{\def\ta@divlist{{}{}{}{}{}{}{}{}{}#1}}
```

`\SetXDIVList` Da mit `\SetDIVList` aus Kompatibilitätsgründen nur Werte für Größen ab 10 pt gesetzt werden können, wird für Werte ab 1 pt ein neuer Befehl benötigt.

```
\newcommand*\SetXDIVList[1]{\def\ta@divlist{#1}}
```

`\ta@bcor` Dimension für die Bindekorrektur reservieren. Diese wird implizit zu 0 pt initialisiert.

```
\newlength{\ta@bcor}
```

`\ta@div` Zähler für *DIV* reservieren. Dieser wird implizit zu 0 initialisiert.

```
\newcount\ta@div
```

Ab hier nun Teile für Klassen und Paket:

```
⟨/package⟩
```

\@BCOR
\@@BCOR

Das Makro \@BCOR wird später benötigt, um die Option BCOR*BCOR* zu erkennen. Die Option wird über ihr Muster erkannt.

```
\newcommand*{\@BCOR}{\expandafter\@@BCOR\CurrentOption BCORBCOR\@@BCOR}
\newcommand*{\@@BCOR}{}
\def\@@BCOR #1BCOR#2BCOR#3\@@BCOR{%
```

Handelt es sich um die Option BCOR, so ist der erste Parameter leer.

```
  \ifstr{#1}{}{%
```

Außerdem ist der dritte Parameter dann BCOR.

```
    \ifstr{#3}{BCOR}{%
```

Der zweite Parameter sollte dann die Größe enthalten. Ist dies nicht der Fall, wird ein Fehler gemeldet (Fehlerzeile ist dann \ProcessOptions).

```
⟨class⟩      \PassOptionsToPackage{BCOR#2}{typearea}%
⟨package⟩      \ifdimen{#2}{\setlength{\ta@bcor}{#2}}{\scr@pti@nerr@r}%
```

In allen anderen Fällen ist es nicht diese Option, dann wird ein Fehler ausgegeben. Stattdessen könnte hier auch auf weitere Optionen getestet werden.

```
    }{\scr@pti@nerr@r}%
  }{\scr@pti@nerr@r}%
}
```

\@DIV
\@@DIV

Die grundsätzliche Funktion ist die gleiche wie bei \@BCOR, \@@BCOR.

```
\newcommand*{\@DIV}{\expandafter\@@DIV\CurrentOption DIVDIV\@@DIV}
\newcommand*{\@@DIV}{}
\def\@@DIV #1DIV#2DIV#3\@@DIV{%
  \edef\@tempa{#1}\ifx\@tempa\@empty
    \edef\@tempa{#3}\def\@tempb{DIV}\ifx\@tempa\@tempb
⟨class⟩      \PassOptionsToPackage{DIV#2}{typearea}%
⟨package⟩      \ifnumber{#2}{\ta@div=#2\relax}{\scr@pti@nerr@r}%
    \else\scr@pti@nerr@r\fi
```

Der Unterschied liegt darin, dass bei einem Fehlschlag der Mustererkennung kein Fehler vorliegt, sondern u. U. noch auf BCOR getestet werden muss.

```
  \else\@BCOR\fi
}
```

\@headlines
\@@headlines

Das Macro \@headlines wird benötigt, um die Option headlines zu erkennen.

```
\newcommand{\@headlines}{%
  \expandafter\@@headlines\CurrentOption headlines\@@headlines%
}
\newcommand*{\@@headlines}{}
\def\@@headlines #1headlines#2\@@headlines{
  \edef\@tempa{#2}\edef\@tempb{headlines}\ifx\@tempa\@tempb
```

```
⟨class⟩     \PassOptionsToPackage{#1headlines}{typearea}
⟨package⟩     \setlength{\@tempdima}{#1\baselineskip}%
⟨package⟩     \def\ta@hh{#1\baselineskip}%
  \else\@DIV\fi
}
```

Jetzt kommt ein Stück, das nur im Paket benötigt wird.

```
⟨*package⟩
```

`\x@paper` `\x@@paper` Diese Makros werden ebenfalls später benötigt, wenn die Papiergröße festgelegt wird.

`\isopaper` Da das Makro `\isopaper` zur Berechnung unterschiedlicher ISO/DIN-Papierformate in der Mustererkennung für die Papiergröße benötigt wird, wurde dieses Makro ebenfalls vorgezogen.

```
\newcommand\isopaper[2][a]{%
```

Das Makro kennt zwei Parameter, von denen der erste und optionale die Grundgröße angibt. Die Angabe erfolgt über einen Buchstaben. Möglich sind

a – ISO/DIN-A-Format

b – ISO/DIN-B-Format

c – ISO/DIN-C-Format

d – ISO/DIN-D-Format

Der zweite Paremeter gibt dann die Skalierung an.

Als Grundgröße wird jeweils die 0-Größe eingesetzt.

```
  \ifx a#1
    \setlength{\paperwidth}{841sp}%
    \setlength{\paperheight}{1189sp}%
  \else\ifx A#1
      \setlength{\paperwidth}{841sp}%
      \setlength{\paperheight}{1189sp}%
    \else\ifx b#1
        \setlength{\paperwidth}{1000sp}%
        \setlength{\paperheight}{1414sp}%
      \else\ifx B#1
          \setlength{\paperwidth}{1000sp}%
          \setlength{\paperheight}{1414sp}%
        \else\ifx c#1
            \setlength{\paperwidth}{917sp}%
            \setlength{\paperheight}{1297sp}%
          \else\ifx C#1
              \setlength{\paperwidth}{917sp}%
              \setlength{\paperheight}{1297sp}%
            \else\ifx d#1
```

```
                \setlength{\paperwidth}{771sp}%
                \setlength{\paperheight}{1090sp}%
              \else\ifx D#1
                  \setlength{\paperwidth}{771sp}%
                  \setlength{\paperheight}{1090sp}%
```

Wurde ein anderer Buchstabe angegeben, ist ein Fehler auszugeben.

```
                \else
                  \PackageError{typearea}{%
                    Unknown ISO/DIN-papersize-letter #1%
                  }{%
                    There are only four defined paper scalings:\MessageBreak
                    a = ISO/DIN-A, b = ISO/DIN-B, c = ISO/DIN-C,
                    d = ISO/DIN-D\MessageBreak
                    Read KOMA-Script-documentation for further information.%
                  }%
                  \setlength{\paperwidth}{841sp}%
                  \setlength{\paperheight}{1189sp}%
                \fi
              \fi
            \fi
          \fi
        \fi
      \fi
    \fi
  \fi
```

Es dürfte aufgefallen sein, dass bei der Größenfestlegung statt der Einheit „mm“ die Einheit „sp“ geschrieben wurde. Die Maße sind also um ein Vielfaches zu klein und müssen später noch mit 186468 multiplitziert werden, um auf Millimeter zu kommen. Dieser Trick ist jedoch notwendig, weil die Norm vorschreibt, dass bei der nun folgenden n-fachen Halbierung der Seiten (n = Skalierung + 1) jeweils auf ganze Millimeter abgerundet werden muss.

```
  \@tempcnta=#2\relax
  \@whilenum\@tempcnta>0
  \do{\setlength{\@tempdima}{\paperwidth}
    \setlength{\paperwidth}{.5\paperheight}
    \setlength{\paperheight}{\@tempdima}
    \advance\@tempcnta by\m@ne
  }
```

Zum Schluß muss nun die genannte Multiplitkation erfolgen. Da „mm“ kein ganzzahliges Vielfaches von „sp“ ist, beinhaltet diese Multiplikation eigentlich einen Fehler und eine Abweichung von der Norm. Da jedoch „sp“ die Rechengenauigkeit von TEX darstellt, stellt auch die Angabe eines ganzzahligen Millimeterwertes bereits einen solchen Fehler dar.

```
  \multiply\paperwidth by186468
  \multiply\paperheight by186468
```

Jetzt noch die gewünschte Ausrichtung beachten.

```
  \if@landscape
    \setlength{\@tempdima}{\paperwidth}%
    \setlength{\paperwidth}{\paperheight}%
    \setlength{\paperheight}{\@tempdima}%
  \fi
}
```

`\iso@paper` Ein ganz ähnlicher interner Befehl erhält den gesamten Namen des ISO-Papierformats als einziges Argument.

`\iso@@paper` Dieses Argument wird dann über einen weiteren internen Befehl per Mustererkennung in den Kennbuchstaben und eine Zahl zerlegt.

```
\newcommand*{\iso@paper}[1]{\expandafter\iso@@paper#1\@nil}
\newcommand*{\iso@@paper}{}
\def\iso@@paper#1#2\@nil{%
  \@tempswafalse
  \if #1a\@tempswatrue
  \else\if #1A\@tempswatrue
    \else\if #1b\@tempswatrue
      \else\if #1B\@tempswatrue
        \else\if #1c\@tempswatrue
          \else\if #1C\@tempswatrue
            \else\if #1d\@tempswatrue
              \else\if #1D\@tempswatrue
  \fi\fi\fi\fi\fi\fi\fi\fi
  \if@tempswa
    \ifnumber{#2}{\isopaper[#1]{#2}}{\@tempswafalse}%
  \fi
}
```

Jetzt wird es wieder für alle interessant:

```
⟨/package⟩
```

Zurück zur Mustererkennung. Diese ist tatsächlich etwas einfacher als die in den beiden Fällen zuvor.

```
\newcommand*{\x@paper}{%
  \expandafter\x@@paper\CurrentOption paper\x@@paper%
}
\newcommand*{\x@@paper}{}
\def\x@@paper #1#2paper#3\x@@paper{%
  \ifstr{#3}{paper}{%
    \ifnumber{#2}{%
      \ifstr{#1}{a}{%
⟨class⟩         \PassOptionsToPackage{#3=#1#2}{typearea}%
⟨package⟩         \isopaper[a]{#2}%
      }{%
        \ifstr{#1}{b}{%
⟨class⟩           \PassOptionsToPackage{#3=#1#2}{typearea}%
```

```
⟨package⟩            \isopaper[b]{#2}%
          }{%
            \ifstr{#1}{c}{%
⟨class⟩              \PassOptionsToPackage{#3=#1#2}{typearea}%
⟨package⟩              \isopaper[c]{#2}%
            }{%
              \ifstr{#1}{d}{%
⟨class⟩                \PassOptionsToPackage{#3=#1#2}{typearea}%
⟨package⟩                \isopaper[d]{#2}%
              }{\scr@pti@nerr@r}%
            }%
          }%
        }%
      }{\scr@pti@nerr@r}%
    }{%
⟨class⟩     \@pt@scan
⟨package⟩     \@headlines
    }%
}
```

`\scr@pti@nerr@r` Falls die zuätzliche Option nicht `BCOR`, `DIV`, `headlines` oder eine bekannte Papiergröße ist, ist die Option nicht verwendet.

```
\newcommand*{\scr@pti@nerr@r}{
⟨class⟩   \OptionNotUsed
⟨package⟩   \@unknownoptionerror
}
```

Jetzt wieder nur für Pakete:

```
⟨*package⟩
```

6.3 Optionen mit keyval

6.3.1 Kopf-, Fuß-, Rand- und Textbereich

`headlines` Neben der Auswertung der ⟨*Wert*⟩`headlines` Form über Pattern Matching wird hier auch noch eine keyval-Option definiert, die bevorzugt zu verwenden ist.

```
\KOMA@key{headlines}{%
  \begingroup
    \setlength{\@tempdima}{#1\baselineskip}%
  \endgroup
  \def\ta@hh{#1\baselineskip}%
}
```

`headheight` Manchmal ist es nützlicher, wenn man die Höhe direkt setzen kann, statt sie über die Anzahl der Grundlinien zu setzen. In dem Fall nimmt man einfach diese Option.

```
\KOMA@key{headheight}{%
```

```
  \begingroup
    \setlength{\@tempdima}{#1}%
  \endgroup
  \def\ta@hh{#1}%
}
```

headinclude headexclude

Diese Optionen bestimmen, ob der Seitenkopf bereits zum Textbereich gehört.

\if@hincl \@hincltrue \@hinclfalse

Der Zustand selbst wird in einem Schalter gespeichert.

```
\newif\if@hincl\@ifundefined{if@hsl}{}{\let\if@hincl\if@hsl}
\KOMA@ifkey{headinclude}{@hincl}
\KOMA@DeclareDeprecatedOption[typearea]{headexclude}{headinclude=false}
```

footinclude footexclude

Diese Optionen bestimmen, ob die Fußzeilen noch zum Textbereich gehören.

\if@fincl \@fincltrue \@finclfalse

Der Zustand selbst wird in einem Schalter gespeichert.

```
\newif\if@fincl\@ifundefined{if@fsl}{}{\let\if@fincl\if@fsl}
\KOMA@ifkey{footinclude}{@fincl}
\KOMA@DeclareDeprecatedOption[typearea]{footexclude}{footinclude=false}
```

mpinclude mpexclude

Es gibt Dokumente mit sehr ausführlichen Randnotizen, bei denen die Randnotizen bei der Satzspiegelberechnung in den Satzspiegel eingerechnet werden sollten. Diese Optionen bestimmen, ob dies der Fall ist.

\if@mincl \@mincltrue \@minclfalse

Der Zustand wird in einem Schalter gespeichert. **Achtung:** Bei Verwendung dieser Option müssen größere DIV-Werte verwendet werden. Siehe dazu weiter unten.

```
\KOMA@ifkey{mpinclude}{@mincl}
\KOMA@DeclareDeprecatedOption[typearea]{mpexclude}{mpinclude=false}
```

BCOR

Der Wert wird einfach an die Länge `\ta@bcor` verfüttert, wobei symbolische Werte speziell behandelt werden, auch wenn sie beim Laden des Pakets absolut keinen Sinn ergeben!

```
\KOMA@key{BCOR}{%
  \edef\@tempa{#1}\edef\@tempb{current}\ifx\@tempa\@tempb\else
    \setlength{\ta@bcor}{#1}%
  \fi
}
```

Wenn allerdings der Satzspiegel bereits berechnet ist, dann muss er hier stattdessen neu berechnet werden.

```
\AtEndOfPackage{\KOMA@key{BCOR}{\typearea[{#1}]{last}}}
```

DIV

Der Wert wird einfach an den Zähler `\ta@div` verfüttert, wobei symbolische Werte speziell behandelt werden. Dabei werden auch die Werte „`last`“ und „`current`“ ausgewertet, auch wenn diese beim Laden des Pakets absolut keinen Sinn ergeben!

```
\KOMA@key{DIV}[calc]{%
  \edef\@tempa{#1}%
  \def\@tempb{last}\ifx\@tempa\@tempb\let\@tempa=\@lastdiv\fi
  \def\@tempb{current}\ifx\@tempa\@tempb\else
    \def\@tempb{default}\ifx\@tempa\@tempb\ta@div=0 \else
      \def\@tempb{calc}\ifx\@tempa\@tempb\ta@div=2 \else
        \def\@tempb{classic}\ifx\@tempa\@tempb\ta@div=3 \else
          \ta@div=\@tempa
        \fi
      \fi
    \fi
  \fi
}
```

Wenn allerdings der Satzspiegel bereits berechnet ist, dann muss er hier stattdessen neu berechnet werden.

```
\AtEndOfPackage{\KOMA@key{DIV}{\typearea[current]{#1}}}
```

`DIVcalc` `DIVclassic`

Dann gibt es da noch zwei alte Sonderfälle.

```
\KOMA@DeclareDeprecatedOption[typearea]{DIVcalc}{DIV=calc}
\KOMA@DeclareDeprecatedOption[typearea]{DIVclassic}{DIV=classic}
```

6.3.2 Sonstige Satzspiegel-Optionen

`oneside` `twoside` `if@semitwoside` `if@twoside`

Einseitiger oder zweiseitiger Satzspiegel?

```
\newif\if@semitwoside
\g@addto@macro\@twosidetrue\@semitwosidefalse
\g@addto@macro\@twosidefalse\@semitwosidefalse
⟨/package⟩
```

```
\KOMA@key{twoside}[true]{%
⟨class⟩  \if@atdocument\else\PassOptionsToPackage{twoside=#1}{typearea}\fi%
⟨*package⟩
  \if@atdocument\if@twoside\cleardoublepage\else\clearpage\fi\fi
  \ifstr{#1}{semi}{%
    \@twosidetrue
    \@semitwosidetrue
  }{%
    \def\FamilyElseValue{, ‘semi’}%
    \KOMA@set@ifkey{twoside}{@twoside}{#1}%
  }%
  \if@twoside
    \if@semitwoside
      \@mparswitchfalse
    \else
      \@mparswitchtrue
    \fi
  \else
    \@mparswitchfalse
  \fi%
```

```
  \csname recalctypearea\endcsname
⟨/package⟩
}
\KOMA@DeclareStandardOption%
⟨package⟩  [typearea]%
  {oneside}{twoside=false}
⟨*package⟩
```

onecolumn Einspaltiger oder zweispaltiger Satz?

twocolumn

```
⟨/package⟩
\KOMA@key{twocolumn}[true]{%
⟨class⟩  \if@atdocument\else\PassOptionsToPackage{twocolumn=#1}{typearea}\fi%
⟨*package⟩
  \if@atdocument\clearpage\fi
  \KOMA@set@ifkey{twocolumn}{@twocolumn}{#1}%
  \csname recalctypearea\endcsname
⟨/package⟩
}
\KOMA@DeclareStandardOption%
⟨package⟩  [typearea]%
  {onecolumn}{twocolumn=false}
⟨*package⟩
```

6.3.3 Ausgabeformat-Optionen

Diese Optionen arbeiten in Abhänigkeit des Ausgabeformats.

pagesize Diese Option ist nun das zentrale Element für die Übermittlung der Seitengröße an pdfTEX, `dvips` oder VTEX. Sie wird auch von den obsoleten Optionen verwendet. Die Werte und ihre Bedeutung:

- `auto` – Kombination aus den Optionen `pdftex` und `dvips`.
- `automedia` – wie `auto` allerdings werden zusätzlich `\mediawidth` und `\mediaheight` gesetzt, falls sie definiert sind.
- `dvips` – das `\special` für `dvips` wird ausgegeben, falls `dvi`-Ausgabe erwartet wird oder `\if@dvips` erfüllt ist. Führt zu einer Warnung, wenn sie mehr als einmal oder nach der Präambel verwendet wird.
- `dvipdfmx` – das `\special` für `dvipdfmx` wird ausgegeben, falls `dvi`-Ausgabe erwartet wird.
- `false` – die automatische Anpassung bei Änderungen innerhalb des Dokuments abschalten.
- `pdftex` – `\pdfpagewidth` und `\pdfpageheight` werden gesetzt, falls sie definiert sind.

```
\KOMA@key{pagesize}[auto]{%
  \begingroup
    \KOMA@set@ncmdkey{pagesize}{@tempa}{%
      {off}{-1},{no}{-1},{false}{-1},%
      {auto}{0},{automedia}{1},{dvips}{2},{pdftex}{3},{dvipdfmx}{4}%
    }{#1}%
    \ifcase\number\@tempa\relax
      \endgroup
      \set@pdftexpagesize
      \set@dvipspagesize
    \or
      \endgroup
      \set@pdftexpagesize
      \set@vtexpagesize
      \set@dvipspagesize
    \or
      \endgroup
      \set@dvipspagesize
    \or
      \endgroup
      \set@pdftexpagesize
    \or
      \endgroup
      \set@dvipdfmxpagesize
  \fi
  \edef\@pagesizelast{#1}%
}
```

\set@pdftexpagesize
\set@dvipspagesize
\setnot@dvipspagesize
\set@dvipdfmxpagesize
\set@vtexpagesize
\scr@updatedvipspagesize

Damit die Option `pagesize` überhaupt funktionieren kann, werden einige interne Makros benötigt.

Achtung: die Definition dieser Makros ändert sich teilweise bei ihrem Aufruf und insgesamt bei `\begin{document}`.

```
\newcommand{\set@pdftexpagesize}{%
  \AtBeginDocument{%
    \ifnotundefined{pdfpagewidth}{\pdfpagewidth=\paperwidth}{}%
    \ifnotundefined{pdfpageheight}{\pdfpageheight=\paperheight}{}%
  }%
  \let\set@pdftexpagesize=\@empty
}
\newcommand{\set@dvipspagesize}{%
  \begingroup
    \@tempswafalse
    \scr@ifdvioutput{\@tempswatrue}{%
      \ifnotundefined{if@dvips}{\let\if@tempswa\if@dvips}{}%
    }%
    \if@tempswa
      \endgroup
      \AtBeginDocument{\AtBeginDvi{%
```

```
          \special{papersize=\the\paperwidth,\the\paperheight}%
          \if@seascape\special{! TeXDict begin /landplus90{true}store end }\fi
        }%
      }%
      \ifnum \scr@compatibility>\@nameuse{scr@v@3.02b}\relax
        \let\set@dvipspagesize=\@empty
      \else
        \let\set@dvipspagesize=\setnot@dvipspagesize
      \fi
    \else
      \endgroup
  \fi
}
\newcommand{\setnot@dvipspagesize}{%
  \scr@ifdvioutput{%
    \PackageWarning{%
      typearea%
    }{%
      With dvi output option ‘pagesize=auto’,\MessageBreak
      ‘pagesize=automedia’ or ‘pagesize=dvips’\MessageBreak
      may be used only once at preamble.\MessageBreak
      Option ignored%
    }%
  }{}%
}
\newcommand*{\set@dvipdfmxpagesize}{%
  \scr@ifdvioutput{%
    \AtBeginDocument{\AtBeginDvi{%
        \special{papersize=\the\paperwidth,\the\paperheight}%
      }%
    }%
  }{}%
}
\newcommand{\set@vtexpagesize}{%
  \AtBeginDocument{%
    \ifnotundefined{mediawith}{\mediawidth=\paperwidth}{}%
    \ifnotundefined{mediaheight}{\mediaheight=\paperheight}{}%
  }%
  \let\set@vtexpagesize=\@empty
}
\newcommand*{\scr@updatedvipspagesize}{%
  \special{ps::\@percentchar\@percentchar
    PageBoundingBox: 0 0 \the\numexpr\paperwidth/\dimexpr 1bp\relax\relax
    \space \the\numexpr\paperheight/\dimexpr 1bp\relax\relax^^J%
    \@percentchar\@percentchar BeginPageSetup^^J%
    << /PageSize [\the\numexpr\paperwidth/\dimexpr 1bp\relax\relax\space
    \the\numexpr\paperheight/\dimexpr 1bp\relax\relax]
    >> setpagedevice^^J%<<
    0 0 bop^^J%
```

```
    \@percentchar\@percentchar EndPageSetup}%
}
\AtBeginDocument{%
  \renewcommand*{\set@pdftexpagesize}{%
    \ifnotundefined{pdfpagewidth}{\pdfpagewidth=\paperwidth}{}%
    \ifnotundefined{pdfpageheight}{\pdfpageheight=\paperheight}{}%
  }%
  \ifnum \scr@compatibility>\@nameuse{scr@v@3.02b}\relax
    \renewcommand*{\set@dvipspagesize}{%
      \scr@ifundefinedorrelax{AtNextShipout}{%
        \scr@ifundefinedorrelax{AtBeginShipoutNext}{%
          \begingroup
            \protected@edef\reserved@a{\endgroup
            \scr@updatedvipspagesize}\reserved@a
        }{%
```

Wenn atbegshi geladen wurde, bzw. `\AtBeginShipoutNext` zur Verfügung steht, dann soll dieser Code noch vor Code ausgeführt werden, der eventuell in `\shipout` eingeschleust wurde. Das funktioniert leider nicht wirklich zuverlässig. Sollte zwischen dem Aufruf von `\set@dvipspagesize` (also der Neuberechnung des Satzspiegels) und der Ausgabe der Seite doch noch etwas per `\AtBeginShipout` oder `\AtBeginShipoutNext` eingefügt werden, dann gehen diese Teile verloren!

```
          \begingroup
            \protected@edef\reserved@a{\endgroup
            \protect\AtBeginShipoutNext{%
              \protect\bgroup
                \protect\global\protect\setbox\AtBeginShipoutBox\protect\vbox{%
                  \scr@updatedvipspagesize
                  \protect\box\AtBeginShipoutBox
                }%
              \protect\egroup
            }}\reserved@a
        }%
      }{%
```

Wenn everyshi geladen wurde, bzw. `\AtNextShipout` zur Verfügung steht, dann soll dieser Code noch vor Code ausgeführt werden, der eventuell in `\shipout` eingeschleust wurde. Das funktioniert leider nicht wirklich zuverlässig. Sollte zwischen dem Aufruf von `\set@dvipspagesize` (also der Neuberechnung des Satzspiegels) und der Ausgabe der Seite doch noch etwas per `\AtNextShipout` oder `\EveryShipout` eingefügt werden, dann gehen diese Teile verloren!

```
        \begingroup
          \protected@edef\reserved@a{\endgroup
          \protect\AtNextShipout{%
            \protect\bgroup
              \protect\global\protect\setbox\protect\@cclv\protect\vbox{%
```

```
                \scr@updatedvipspagesize
                \protect\box\protect\@cclv
              }%
            \protect\egroup
          }}\reserved@a
      }%
    }%
  \else
    \let\set@dvipspagesize=\setnot@dvipspagesize
  \fi
  \renewcommand*{\set@dvipdfmxpagesize}{%
    \scr@ifdvioutput{%
      \special{papersize=\the\paperwidth,\the\paperheight}%
    }{}%
  }%
  \renewcommand*{\set@vtexpagesize}{%
    \ifnotundefined{mediawith}{\mediawidth=\paperwidth}{}%
    \ifnotundefined{mediaheight}{\mediaheight=\paperheight}{}%
  }%
}
```

`dvips` Die Option ist obsolete und wird nun über die Option `pagesize` implementiert. Obwohl die Verwendung von `pagesize=automedia` zu einer Warnung führen kann, wird hier nur informiert.

```
\KOMA@DeclareStandardOption[typearea]{dvips}{%
  pagesize=automedia%
}
```

`pdftex` Die Option ist obsolete und wird nun über die Option `pagesize` implementiert.

```
\KOMA@DeclareStandardOption[typearea]{pdftex}{%
  pagesize=pdftex%
}
```

6.3.4 Papiergrößen-Optionen

Die folgende Option bestimmt die Papiergröße und -ausrichtung. Aus der Papiergröße wird später mit `\typearea` der Satzspiegel berechnet. Darüber hinaus wäre das eine passende Gelegenheit `\SetDIVList` anzuwenden, darauf wird aber verzichtet, da die Verwendung der Liste ohnehin als veraltete Methode betrachtet werden kann.

`a4paper`
`a5paper`
`b5paper`
`letterpaper`
`legalpaper`
`executivepaper`
`landscape`
`portrait`

Die Formate, die auch bei den Standardklassen definiert sind, werden als Sonderfall explizit definiert. Dadurch ist ein Wechsel der Klasse bzw. die Verwendung von typearea mit den Standardklassen einfacher. Aus Kompatibilitätsgründen werden auch die übrigen Papiergrößenoptionen später noch per Mustererkennung verarbeitet.

```
\KOMA@DeclareStandardOption[typearea]{a4paper}{paper=a4}
\KOMA@DeclareStandardOption[typearea]{a5paper}{paper=a5}
\KOMA@DeclareStandardOption[typearea]{b5paper}{paper=b5}
\KOMA@DeclareStandardOption[typearea]{letterpaper}{paper=letter}
\KOMA@DeclareStandardOption[typearea]{legalpaper}{paper=legal}
\KOMA@DeclareStandardOption[typearea]{executivepaper}{paper=executive}
\KOMA@DeclareStandardOption[typearea]{landscape}{paper=landscape}
\KOMA@DeclareStandardOption[typearea]{portrait}{paper=portrait}
```

`paper` Im Normalfall wird die Papiergröße jedoch über diese Option gesetzt. Dabei sind neben den Standard- und den ISO-Reihen A, B, C und D auch Wertepaare der Form „`(`⟨*Breite*⟩`,`⟨*Höhe*⟩`)`“ und die Angaben „`portrait`“ und „`landscape`“ erlaubt.

```
\KOMA@key{paper}{%
  \ifstr{#1}{seascape}{%
    \@seascapetrue
    \if@landscape\else
      \setlength{\@tempdima}{\paperwidth}%
      \setlength{\paperwidth}{\paperheight}%
      \setlength{\paperheight}{\@tempdima}%
      \@landscapetrue
    \fi
  }{%
    \ifstr{#1}{landscape}{%
      \@seascapefalse
      \if@landscape\else
        \setlength{\@tempdima}{\paperwidth}%
        \setlength{\paperwidth}{\paperheight}%
        \setlength{\paperheight}{\@tempdima}%
        \@landscapetrue
      \fi
    }{%
      \ifstr{#1}{portrait}{%
        \if@landscape
          \setlength{\@tempdima}{\paperwidth}%
          \setlength{\paperwidth}{\paperheight}%
          \setlength{\paperheight}{\@tempdima}%
          \@landscapefalse
        \fi
      }{%
        \ifstr{#1}{letter}{%
          \if@landscape
            \setlength\paperheight{8.5in}%
            \setlength\paperwidth {11in}%
          \else
            \setlength\paperheight{11in}%
            \setlength\paperwidth {8.5in}%
          \fi
        }{%
```

```
          \ifstr{#1}{legal}{%
            \if@landscape
              \setlength\paperheight{8.5in}%
              \setlength\paperwidth {14in}%
            \else
              \setlength\paperheight{14in}%
              \setlength\paperwidth {8.5in}%
            \fi
          }{%
            \ifstr{#1}{executive}{%
              \if@landscape
                \setlength\paperheight{7.25in}%
                \setlength\paperwidth {10.5in}%
              \else
                \setlength\paperheight{10.5in}%
                \setlength\paperwidth {7.25in}%
             \fi
            }{%
              \iso@paper{#1}%
              \if@tempswa\else
                \expandafter\maybe@pair#1::\@nil
                \if@tempswa
                  \ifnum \scr@ta@compatibility<\@nameuse{scr@v@3.01b}\relax
                    \PackageWarningNoLine{typearea}{%
                      You may have to exchange height and width!\MessageBreak
                      Until KOMA-Script version 3.01b the semantic\MessageBreak
                      of option ‘paper’ was ‘paper=height:width’\MessageBreak
                      instead of ‘paper=width:height’.\MessageBreak
                      I’ve switched back to that old interpretation\MessageBreak
                      because of the given compatiblity settings}%
                    \setlength\paperheight{\@tempa}%
                    \setlength\paperwidth{\@tempb}%
                  \else
                    \setlength\paperwidth{\@tempa}%
                    \setlength\paperheight{\@tempb}%
                  \fi
                \else
                  \KOMA@unknown@keyval{paper}{#1}{%
                    ‘letter’, ‘legal’, ‘executive’, ‘a’<nr>, ‘b’<nr>,
                    ‘c’<nr>, ‘d’<nr>,\MessageBreak
                    <width>‘:’<height>, ‘landscape’ and ‘portrait’%
                  }%
                \fi
              \fi
            }%
          }%
        }%
      }%
    }%
```

```
  }%
}
```

\if@seascape
\@seascapetrue
\@seascapefalse

In diesem Schalter wird nur gespeichert, ob Breitformat ggf. umgekehrt verwendet werden soll. Dies macht derzeit nur bei Verwendung von dvips einen Unterschied!

```
\newif\if@seascape
```

\if@landscape
\@landscapetrue
\@landscapefalse

In diesem Schalter wird gespeichert, ob Hoch- oder Breitformat verwendet wird.

```
\newif\if@landscape
```

\maybe@pair

Testet, ob der nachfolgende Parameter die Form „⟨*erstes Argument*⟩:⟨*zweites Argument*⟩“ hat. Wenn ja, wird `\@tempa` mit dem ersten und `\@tempb` mit dem zweiten Argument definiert.

```
\newcommand*{\maybe@pair}{}
\def\maybe@pair#1:#2:#3\@nil{%
  \edef\@tempa{#1}\edef\@tempb{#2}%
  \ifx\@tempa\@empty\@tempswafalse\else
    \ifx\@tempb\@empty\@tempswafalse\else\@tempswatrue\fi
  \fi
}
```

```
⟨/package⟩
```

6.3.5 Geparste Optionen

Die Behandlung für `BCOR`*Wert*, `DIV`*Wert* und `a`*Wert*`paper`, `b`*Wert*`paper`, `c`*Wert*`paper` sowie `d`*Wert*`paper` und *Wert*`headlines`.

```
\DeclareOption*{%
⟨letter⟩  \lco@test
  \x@paper}
```

6.3.6 Konfigurationsdatei

Es ist möglich in einer Datei `typearea.cfg` z. B. weitere Optionen zu definieren, die dann automatisch eingeladen werden.

```
\InputIfFileExists{typearea.cfg}{%
  \typeout{%
    ****************************************^^J%
    * Local config file typearea.cfg used^^J%
    ****************************************%
  }
}{}
```

6.3.7 Standardoptionen

Als Standardoptionen sind `headinclude=false`, `footinclude=false` und die Papiergröße `paper=A4` eingestellt. Diese Optionen können selbstverständlich durch explizite Angaben überladen werden.

```
\KOMAExecuteOptions{paper=A4}
\if@twoside
  \if@semitwoside
    \KOMAExecuteOptions{twoside=semi}
  \else
    \KOMAExecuteOptions{twoside}
  \fi
\else
  \KOMAExecuteOptions{twoside=false}
\fi
\KOMAProcessOptions\relax
```

6.4 Schriftgrößenabhängige Voreinstellungen für DIV

`\ta@divlist`
`\ta@divlistelse`

An dieser Stelle muss noch für die Standardeinstellung der *DIV*-Liste gesorgt werden, falls eine solche noch nicht definiert wurde.

```
\providecommand\ta@divlist{{}{}{}{}{}{}{}{}{}{8}{10}{12}}
```

Wenn kein Defaultwert definiert ist, wird dieser Wert verwendet:

```
\providecommand\ta@divlistelse{1}
```

Das waren die Optionen für das Paket oder im ersten Durchlauf der Klassen.

⟨/package⟩

6.5 Laden des Pakets

Das Folgende wird nur in einem zweiten Durchlauf in die Klassen eingefügt und sorgt für das Laden des typearea Pakets.

⟨*class⟩

Vor dem Laden müssen noch diverse Makros zurückgesetzt werden, damit sie überhaupt in typearea definiert werden können:

```
\let\@BCOR                =\relax
\let\@@BCOR               =\relax
\let\@DIV                 =\relax
\let\@@DIV                =\relax
\let\@headlines           =\relax
\let\@@headlines          =\relax
\let\x@paper              =\relax
\let\x@@paper             =\relax
\let\scr@pti@nerr@r       =\relax
```

Jetzt kann geladen werden: Die Überprüfung auf die korrekte KOMA-Script-Version geschieht in dem Paket (siehe dazu `scrkvers.dtx`).

```
\RequirePackage{typearea}[\KOMAScriptVersion]
```

Auf diese Art ist sichergestellt, dass immer die Version von typearea verlangt wird, die zur gleichen Release wie die Klassen gehört. Wobei die Warnungen je nach Zusammentreffen der Versionen unterschiedlich ausfallen können.

```
⟨/class⟩
```

Ab hier ist alles nur noch für das Paket.

```
⟨*package⟩
```

6.6 Einige interne Register

`\ta@hblk` `\ta@vblk` `\ta@temp`

Diese Größsenparameter und werden intern für verschiedene Befehle benötigt. (Die Zähler `\ta@bcor` und `\ta@div` werden bereits bei der Optionenauswertung benötigt und sind deshalb vorgezogen.)

```
\newlength{\ta@hblk}
\newlength{\ta@vblk}
\newlength{\ta@temp}
```

6.7 Die Berechnung des Satzspiegels

`\typearea`

Die Berechnung des Satzspiegels, wie sie in der Anleitung beschrieben ist, findet mit Hilfe des Befehls `\typearea` statt, der hier definiert wird. Die Berechnung selbst benötigt mehreren Stufen.

`\@typearea`

Die eigentliche Satzspiegelberechnung geschieht über das Hilfsmakro `\@typearea`, das dann auch von `\typearea` genutzt wird. Dieses Hilfsmakro benötigt keine Parameter, da diese bereits zuvor von `\typearea` in die entsprechenden Variablen geschrieben wurden.

```
\newcommand*\@typearea{%
```

Bevor irgend etwas anderes gemacht wird, sogen wir dafür, dass die gewählte Schrift auch gesetzt wird. Dies ist notwendig, falls jemand vor `\typearea` einen Befehl wie `\linespread` verwendet.

```
  \selectfont
```

Zunächst werden die horizontalen Maße berechnet. Dazu wird als erstes die Größe einer horizontalen Längeneinheit (*HLE*) ermittelt:

```
  \ta@hblk=\paperwidth
  \advance \ta@hblk by-\ta@bcor
  \divide\ta@hblk by\ta@div
```

Die Größe des Bereichs für die Randnotizen ist ebenfalls von der Streifenbreite abhängig.

```
  \if@mincl
    \marginparwidth=\ta@hblk
```

```
    \advance\marginparwidth by-\marginparsep
  \else
    \if@twoside
      \if@semitwoside
        \marginparwidth=\ta@hblk
      \else
        \marginparwidth=1.5\ta@hblk
      \fi
    \else
      \marginparwidth=\ta@hblk
    \fi
  \fi
```

Der linke und rechte Rand wird relativ zur benutzbaren Seitenbreite berechnet. Dabei muss auch der Rand berücksichtigt werden, der vom Druckertreiber standardmäßig hinzugefügt wird. Außerdem ist der Rand abhängig davon, ob es sich um ein einseitiges oder beidseitiges Layout handelt. Bei Option `mpinclude` sind im beidseitigen Satz zusätzlich die Randnotizen zu beachten.

```
  \oddsidemargin=-1in
  \evensidemargin=-1in
  \if@twoside
    \if@semitwoside
      \advance\oddsidemargin by1.5\ta@hblk
      \advance\oddsidemargin by\ta@bcor
      \evensidemargin=\oddsidemargin
    \else
      \advance\oddsidemargin by\ta@hblk
      \advance\oddsidemargin by\ta@bcor
      \advance\evensidemargin by2\ta@hblk
      \if@mincl
        \advance\evensidemargin by\marginparwidth
        \advance\evensidemargin by\marginparsep
      \fi
    \fi
  \else
    \advance\oddsidemargin by1.5\ta@hblk
    \advance\oddsidemargin by\ta@bcor
    \evensidemargin=\oddsidemargin
  \fi
```

Der Rest steht nun für den Textbereich zur Verfügung:

```
  \textwidth=\paperwidth
  \advance\textwidth by-\ta@bcor
  \advance\textwidth by-3\ta@hblk
```

Falls die Randnotizen für die Satzspiegelberechnung zum Textbereich gehören, müssen sie zusammen mit ihrem Abstand vom Text nun von diesem abgezogen werden. Es wird dabei davon ausgegangen, dass die Randnotizen immer außen gesetzt werden. Sollte der Textbereich für Randnotizen

zu schmal sein, so wird der Bereich der Randnotizen so lange verkleinert, bis dies nicht mehr der Fall ist. Am einfachsten geht dies, indem gegebenfalls die Breiten vertauscht werden.

```
\if@mincl
  \advance\textwidth by-\marginparwidth
  \advance\textwidth by-\marginparsep
  \ifdim\textwidth <\marginparwidth
    \PackageWarningNoLine{typearea}{%
      Width of text (\the\textwidth) less than width of\MessageBreak
      marginpar (\the\marginparwidth) while using option\MessageBreak
      `mpinclude'.\MessageBreak
      Because of this, I'm exchanging the values.\MessageBreak
      You should increase DIV to avoid such problems.\MessageBreak
    }%
    \@tempskipa=\marginparwidth
    \marginparwidth=\textwidth
    \textwidth=\@tempskipa
  \fi
\fi
```

Danach folgen die vertikalen Maße. Auch dafür wird zunächst die Höhe eines Streifens benötigt.

```
\ta@vblk=\paperheight
\divide\ta@vblk by\ta@div
```

Davon ausgehend kann der obere Rand berechnet werden, wobei zu berücksichtigen ist, dass TEX diesen selbständig um ein Inch vergrößert.

```
\topmargin=-1in
\advance\topmargin by\ta@vblk
```

Außerdem ist zu berücksichtigen, ob der Kopf zum Textbereich oder zum Rand gehört.

```
\if@hincl \else
  \advance\topmargin by-\headheight
  \advance\topmargin by-\headsep
\fi
```

Bei der Berechnung des Textbereichs sind von der Papierhöhe die Ränder abzuziehen und es ist ebenfalls zu berücksichtigen, ob Kopf- und Fuß zu den Rändern oder zum Textbereich selbst gezählt werden.

```
\ta@temp=\paperheight
\advance\ta@temp by-3\ta@vblk
\if@hincl
  \advance\ta@temp by-\headheight
  \advance\ta@temp by-\headsep
\fi
\if@fincl
  \advance\ta@temp by-\footskip
\fi
```

Als Sonderbedingung ist dann noch zu beachten, dass die Größe des Textbereichs nicht zu einer vertikalen Aufweitung führt.

```
  \scr@ifundefinedorrelax{@ptsize}{%
    \topskip=\f@size\p@
  }{%
    \topskip=\@ptsize\p@
    \advance\topskip by 10\p@
  }%
  \maxdepth=.5\topskip
  \textheight=\topskip
  \@whiledim\textheight <\ta@temp
  \do{\advance\textheight by\baselineskip}%
}
```

`\ta@modiv` Dieses Hilfsmakro wird weiter unten benötigt. Es dividiert den ersten Parameter durch den zweiten und liefert das gerundete Ergebnis in `\ta@div` zurück. Für die Berechnung werden `\@tempdima` und `\@tempdimb` verwendet.

```
\newcommand*\ta@modiv[2]{
  \setlength{\@tempdima}{#1}
  \divide\@tempdima by#2
  \ta@div\@tempdima
  \ifnum \ta@div<4
    \ta@div=4
  \else
    \setlength{\@tempdimb}{#1}%
    \divide\@tempdimb by\ta@div
    \addtolength{\@tempdima}{\@tempdima}%
    \addtolength{\@tempdima}{-\@tempdimb}%
    \advance\ta@div by\@ne
    \setlength{\@tempdimb}{#1}%
    \divide\@tempdimb by\ta@div

    \ifdim \@tempdima<\@tempdimb
      \advance\ta@div by\m@ne
    \fi
  \fi
}
```

`\ta@temp@goodwidth` Dieses Makro setzt die Hilfdimension `\ta@temp` auf den Wert einer optimalen Zeilenbreite in der aktuellen Schrift. Voraussetzung für diese Berechnung ist, dass Zeilenbreiten von ca. 66 bis 72 Zeichen optimal sind. Zur Vereinfachung wird mit dem Mittelwert 69 gerechnet. Eine solche Zeile besteht dann aus 66 durchschnittlichen Klein- und 3 durchschnittlichen Großbuchstaben.

```
\newcommand*\ta@temp@goodwidth{%
  \settowidth{\ta@temp}{\normalfont\normalsize
    abcdefghijklmnopqrstuvwxyz}%
```

```
  \ifdim\ta@temp >200\p@
    \setlength{\ta@temp}{2.53846\ta@temp}%
    \settowidth{\@tempdima}{\normalfont\normalsize
      ABCDEFGHIJKLMNOPQRSTUVWXYZ}%
    \setlength{\@tempdima}{.11538\@tempdima}%
    \addtolength{\ta@temp}{\@tempdima}%
  \else
    \multiply\ta@temp by66
    \settowidth{\@tempdima}{\normalfont\normalsize
      ABCDEFGHIJKLMNOPQRSTUVWXYZ}%
    \multiply\@tempdima by3
    \addtolength{\ta@temp}{\@tempdima}%
    \divide\ta@temp by26
  \fi
}
```

`\AfterCalculatingTypearea` Neuerdings besteht die Möglichkeit, auf Benutzerebene das Verhalten von `\typearea` zu beeinflussen. In der Sternvariante wird dabei lokal gearbeitet.

```
\newcommand*{\AfterCalculatingTypearea}{%
  \@ifstar \l@addto@macro \g@addto@macro \@typearea@end
}
```

`\@typearea@end` Das Makro `\@typearea@end` dient als Hook, um am Ende von `\typearea` noch irgendwelche Änderungen vornehmen zu können. Dies geschieht ggf. noch vor der Aktivierung der Einstellungen innerhalb eines Dokuments.

```
\newcommand*{\@typearea@end}{}
```

`\activateareas` ieses Makro wird verwendet, um mit einem gewaltigen Hack die Änderung des Satzspiegels innerhalb eines Dokuments zu ermöglichen. Es stellt selbst sicher, dass es nur innerhalb des Dokuments ausgeführt wird.

```
\newcommand*{\activateareas}{%
```

Als erstes wird getestet, ob wir überhaupt innerhalb des Dokuments sind und nur dann wird etwas getan.

```
  \if@atdocument
```

Wenn dem so ist, wird jedoch eine Warnung ausgegeben, weil es wirklich sehr unsauber ist, so zu verfahren. Das ganze ist absolut abhängig von der LaTeX-Version.

```
    \PackageWarning{typearea}{%
      Typearea changed!\MessageBreak
      You should do this only at preamble, because only\MessageBreak
      \protect\begin{document} calculates output dimensions!\MessageBreak
      Trying to calculate new output dimensions, but\MessageBreak
      this is only a dirty hack}%
```

Dann können endlich die Dimensionen, die z. B. von der Ausgaberoutine benötigt werden, neu berechnet werden.

```
    \@colht=\textheight
```

```
    \@colroom=\textheight
    \vsize=\textheight
    \columnwidth=\textwidth
    \if@twocolumn
      \advance\columnwidth by-\columnsep
      \divide\columnwidth by\tw@
      \hsize=\columnwidth
      \@firstcolumntrue
    \fi
    \hsize=\columnwidth
    \linewidth\hsize
    \scr@ifundefinedorrelax{@pagesizelast}{}{\KOMAoptions{pagesize=\@pagesizelast
  \fi
}
```

`\ta@divfor` Der Standardseitenspiegel ist abhängig von den mit `\SetDIVList` vorgegebenen Einstellungen. Dazu muss aus einer eventuell vorhandenen inneren Liste `\ta@divlist` der von der Schriftgröße abhängige *DIV*-Wert ausgelesen werden. Dies geschieht mit Hilfe des `\ta@divfor`-Befehls, der sich wiederum einiger interner Befehle zur Listenbearbeitung bedient.

```
\newcommand*{\ta@divfor}[1]{%
  \if@mincl
    \PackageInfo{typearea}{%
      With option ‘mpinclude’ there are no predefined\MessageBreak
      DIV values. Using DIV calculation for good line\MessageBreak
      width%
    }%
    \ta@div=2
  \else
    \xdef\@tempa{\ta@divlist{}}%
    \@tempdima=#1pt
    \addtolength{\@tempdima}{.5\p@}%
    \divide\@tempdima by\p@
    \setlength{\@tempdimb}{\@tempdima}%
    \multiply\@tempdimb by\p@
    \ifdim \@tempdimb=#1pt
      \ta@div=#1\relax
      \@whilenum \ta@div >-9 \do {%
        \ifx\@tempa\empty
        \else \xdef\@tempa{\expandafter\@cdr\@tempa\@nil}\fi
        \advance\ta@div by-1
      }%
    \else
      \let\@tempa\empty
    \fi
    \ifx\@tempa\empty
```

Wenn keine Voreinstellung gefunden werden konnte, so wird immer der Optimalwert berechnet. Mit einer Warnung wird darauf hingewiesen.

```
      \setlength{\@tempdima}{#1\p@}%
      \addtolength{\@tempdima}{10\p@}%
      \PackageWarningNoLine{typearea}{%
        DIV for \the\@tempdima\space and used papersize\MessageBreak
        not defined!\MessageBreak
        Using DIV=\ta@divlistelse
      }%
      \ta@div=\ta@divlistelse\relax
    \else
```

Ansonsten wird der vorgegebene Wert verwendet.

```
      \xdef\@tempa{\expandafter\@car\@tempa\@nil}%
      \ifx\@tempa\empty
        \ta@div=\ta@divlistelse\relax
      \else
        \ta@div=\@tempa\relax
      \fi
    \fi
  \fi
}
```

Die symbolischen Werte für das obligatorische Argument von `\typearea` haben folgende Bedeutung:

Symbol	Wert	Bedeutung
`calc`	2	guten Wert berechnen
`classic`	3	mittelalterlicher Buchseitenkanon
`current`	—	aktuellen Wert beibehalten
`default`	0	Wert aus der Liste verwenden
`last`	—	Argument des letzten Aufrufs verwenden
`areaset`	—	Satzspiegel beibehalten nur neu anordnen

Das optionale Argument kennt hingegen nur den symbolischen Wert `current`.

```
\newcommand*\typearea[2][\z@]{%
  \scr@ifundefinedorrelax{currentgrouptype}{}{%
    \ifcase\currentgrouptype\else
      \PackageWarning{typearea}{%
        \string\typearea\space used at group level
        \the\currentgrouplevel.\MessageBreak
        Using \string\typearea\space inside any group, e.g.\MessageBreak
        environments, math mode, boxes, etc. may result in\MessageBreak
        many type setting problems. Because of this you\MessageBreak
        should never do this. You should move the
        \string\typearea\MessageBreak
        outside all groups%
      }%
    \fi
  }%
```

```
\edef\@tempa{#1}\edef\@tempb{current}\ifx\@tempa\@tempb\else
  \setlength{\ta@bcor}{#1}%
\fi
\edef\@tempa{#2}%
\def\@tempb{last}\ifx\@tempa\@tempb\let\@tempa=\@lastdiv
  \ifx\csname @tempa\endcsname\relax
    \def\@tempa{current}%
  \fi
\fi
\def\@tempb{areaset}\ifx\@tempa\@tempb
  \areaset[current]{\textwidth}{\textheight}%
\else
  \def\@tempb{current}\ifx\@tempa\@tempb\else
    \def\@tempb{default}\ifx\@tempa\@tempb\ta@div=0 \else
      \def\@tempb{calc}\ifx\@tempa\@tempb\ta@div=2 \else
        \def\@tempb{classic}\ifx\@tempa\@tempb\ta@div=3 \else
          \ta@div=\@tempa
        \fi
      \fi
    \fi
  \fi
  \edef\@lastdiv{\the\ta@div}%
```

Bevor nun irgendeine Berechnung durchgeführt wird, muss ggf. eine neue Seite begonnen werden. Außerdem wird sichergestellt, dass die aktuelle Schriftgröße die Grundschriftgröße ist und auch mit dem aktuellen Durchschuss gerechnet wird.

```
  \if@atdocument\newpage\fi
  \normalsize
```

Nun muss festgestellt werden, ob die Höhe des Kopfbereichs bereits vorgegeben ist, oder ob diese selbst berechnet werden soll. Im Falle der Berechnung wird als Höhe für den Kopf das 1,25-Fache des Grundzeilenabstandes gewählt. Damit sollte auch noch genug Raum für die optionale Trennlinie vorhanden sein.

```
  \@ifundefined{ta@hh}{%
    \headheight=1.25\baselineskip
  }{%
    \headheight=\ta@hh
  }%
```

Auch der Abstand zwischen Kopf und Text ist immer gleich. Verschiedene Typographen geben hier verschiedene Werte an. Ein guter Wert sollte erreicht werden, wenn dadurch eine Kopfzeile in das normale Zeilenraster eingepasst wird. Der Abstand soll aber auch nicht zu groß, werden. Optimal ist angeblich ein Wert von einer Zeile. Dies führt jedoch zu einem klaren Konflikt bei unterstrichenen Kopfzeilen. Deshalb wird ein Kompromiss gebildet.

```
  \headsep=1.5\baselineskip
```

Es folgen die restlichen von *DIV* und *BCOR* unabhängigen Werte.

```
\footskip=3.5\baselineskip
\marginparsep=1cc
\marginparpush=0.45\baselineskip
```

Vor der eigentlichen Berechnung des Satzspiegels wird nun zunächst überprüft, ob der übergebene *DIV*-Wert 0 ist. Da Werte von 0 bis 3 nicht möglich sind (Minimum sind drei Teile Rand und ein Teil Text), wird der Wert 0 dafür verwendet, den Defaultwert aus der Liste zu ermitteln, falls ein solcher vorhanden ist.

```
\ifnum\ta@div=\z@
  \@ifundefined{@ptsize}{%
    \@tempdima=\f@size\p@
    \advance\@tempdima by.5pt
    \ta@div=\@tempdima
    \divide\ta@div by65536
    \advance\ta@div by-10
    \xdef\@ptsize{\the\ta@div}%
    \PackageWarningNoLine{typearea}{%
      \string\@ptsize\space not defined!\MessageBreak
      Fontsize is \f@size pt.\MessageBreak
      Defining \string\@ptsize\space to \@ptsize%
    }%
  }{}%
  \ta@divfor{\@ptsize}%
\fi
```

Als zweiter Sonderfall wird für einen *DIV*-Wert von 3 die *Kreisberechnung* durchgeführt. Dabei wird versucht, den Textbereich genau so hoch zu wählen, wie das Papier breit ist. Ein solches Verhältnis wird bei klassischen Papierformaten als harmonisch betrachtet.

```
\ifnum\ta@div=3
  \typeout{DIV calculation for classic typearea.}%
  \setlength{\ta@temp}{\paperwidth}%
  \addtolength{\ta@temp}{-\ta@bcor}%
```

Bei der Berechnung des vertikal benötigten Platzes, ist zu beachten, ob Kopf und Fuß zum Textbereich oder zu den Rändern gehören.

```
  \if@hincl\else
    \addtolength{\ta@temp}{\headheight}%
    \addtolength{\ta@temp}{\headsep}%
  \fi
  \if@fincl\else
    \addtolength{\ta@temp}{\footskip}%
  \fi
```

Wird vertikal mehr Platz benötigt, als vorhanden ist, wird eine andere Berechnung versucht, die weiter unten erfolgt.

```
  \ifdim \ta@temp >\paperheight
    \PackageWarningNoLine{typearea}{%
```

```
          No classic typearea with this papersize!\MessageBreak
          Using calculated DIV%
        }%
        \ta@div=\@ne
```

Ansonsten wird mittels des belegten Textbereichs und der Papierhöhe der obere und untere Rand zusammen,

```
      \else
        \setlength{\ta@temp}{\paperheight}%
        \addtolength{\ta@temp}{-\paperwidth}%
        \addtolength{\ta@temp}{\ta@bcor}%
```

daraus die Streifenhöhe

```
        \divide\ta@temp by3
```

und schließlich erneut mit Hilfe der Papierhöhe der genaue *DIV*-Wert berechnet.

```
        \ta@modiv{\paperheight}{\ta@temp}%
      \fi
```

An dieser Stelle muss `mpinclude` nicht berücksichtigt werden, da nicht die Breite des Textbereichs, sondern die Breite des Papiers als Kriterium verwendet wird.

```
    \fi
```

Wurde bisher kein Wert gefunden oder ein *DIV*-Wert von 1 bis 2 angegeben, so wird nun ein möglichst optimaler *DIV*-Wert ermittelt.

```
    \ifnum\ta@div <4
      \typeout{DIV calculation for typearea with good linewidth.}%
```

Zunächst könnte die Textbreite die gesamte Sichtbare Papierbreite einnehmen:

```
      \setlength{\textwidth}{\paperwidth}%
      \addtolength{\textwidth}{-\ta@bcor}%
```

Als Sollbreite wird hier die ideale Zeilenfüllung verwendet:

```
      \ta@temp@goodwidth
```

Im Zweispaltenmodus besteht jede Zeile aus zwei solchen Spalten zuzüglich dem Spaltenabstand:

```
      \if@twocolumn
        \addtolength{\ta@temp}{\ta@temp}%
        \addtolength{\ta@temp}{\columnsep}%
      \fi
```

In früheren Versionen erfolgte nun eine iterative Annäherung an den optimalen *DIV*-Wert. Je nach TEX-Implementierung und Papierformat, kann das aber recht lange dauern. Die obere Berechnung zum Vorbild nehmend, wird deshalb nun ebenfalls versucht, den optimalen Wert direkt zu berechnen. Dazu wird zunächst wie oben die Größe der horizontalen Ränder zusammen ermittelt:

```
    \setlength{\ta@hblk}{\paperwidth}%
    \addtolength{\ta@hblk}{-\ta@temp}%
```

Daraus dann die Breite eines Streifens.

```
    \divide\ta@hblk by3
```

Ist die Breite negativ, so wird die Mindestbreite von 5mm eingesetzt.

```
    \ifdim \ta@hblk <\z@
      \setlength{\ta@hblk}{5mm}%
    \fi
```

Nun kann wieder der *DIV*-Wert ermittelt werden. Im Falle der Option `mpinclude` gilt allgemein, dass der Nenner mit $\frac{3}{3+m}$ zu multiplizieren ist. Dabei ist m die Anzahl an Längeneinheiten, die für die Randnotizen verwendet werden sollen. Mit $m = 1$ ist also der Faktor $\frac{3}{4} = 0.75$ zu verwenden. Schwieriger wird es, wenn hier nicht die Anzahl Längeneinheiten, sondern ein konkreter Wert für die Breite der Randnotizen verwendet wird.

```
    \if@mincl
      \setlength{\ta@temp}{.75\ta@hblk}%
    \else
      \setlength{\ta@temp}{\ta@hblk}%
    \fi
    \ta@modiv{\paperwidth}{\ta@temp}%
```

Als Erschwernis kommt jetzt jedoch hinzu, dass nun zwar der *DIV*-Wert für die Breite optimal ist, jedoch u. U. zu einer zu großen Höhe führt. Deshalb wird zunächst ein Satzspiegel für diesen *DIV*-Wert berechnet.

```
    \@typearea
```

Nun wird überprüft, ob die vertikalen Ausmaße in Ordnung sind. Auch hier ist wieder eine minimale Streifenbreite vorteilhaft. Die Streifenbreite ergibt sich direkt aus dem oberen Rand. Wobei jedoch noch das TEX-Offset von 1 in zu berücksichtigen ist.

```
    \setlength{\ta@temp}{-1in}%
    \addtolength{\ta@temp}{5mm}%
    \ifdim\topmargin <\ta@temp
```

Ist der Minimalrand nicht eingehalten, so wird für diesen ein neuer *DIV*-Wert ermittelt. Dabei muss ebenfalls berücksichtigt werden, ob Kopf- und Fußbereich zum Text oder zum Rand gehören.

```
      \setlength{\ta@temp}{15mm}%
      \if@hincl\else
        \addtolength{\ta@temp}{\headheight}%
        \addtolength{\ta@temp}{\headsep}%
      \fi
      \if@fincl\else
        \addtolength{\ta@temp}{\footskip}%
      \fi
      \divide\ta@temp by3
      \ta@modiv{\paperheight}{\ta@temp}%
    \fi
```

Dieser Sonderfall ist somit abschließend behandelt.

```
\fi
```

Damit steht jetzt also der *DIV*-Wert fest, und die eigentliche Berechnung kann folgen.

```
\@typearea
```

Zum Schluss soll die Seitenaufteilung noch in die `log`-Datei geschrieben werden.

```
\ta@temp@goodwidth
\if@twocolumn
  \addtolength{\ta@temp}{\ta@temp}%
  \addtolength{\ta@temp}{\columnsep}%
\fi
\setlength{\@tempdima}{\ta@temp}%
\addtolength{\ta@temp}{-\textwidth}%
\divide\@tempdima by100
\divide\ta@temp by\@tempdima
\@tempcnta=\ta@temp
\ifnum\@tempcnta <5
  \ifnum\@tempcnta >-5
    \@tempcnta=0
  \else
    \advance\@tempcnta by4
  \fi
\else
  \advance\@tempcnta by-4
\fi
\ifnum\@tempcnta <-15
  \@tempcnta=-\@tempcnta
  \PackageWarningNoLine{typearea}{%
    Bad type area settings!\MessageBreak
    The detected line width is about
    \the\@tempcnta\@percentchar\MessageBreak
    larger than the heuristically detected line width.\MessageBreak
    You should e.g. decrease DIV, increase fontsize\MessageBreak
    or change papersize%
  }%
  \@tempcnta=-\@tempcnta
\else
  \ifnum\@tempcnta>15
    \PackageWarningNoLine{typearea}{%
      Maybe no optimal type area settings!\MessageBreak
      The maximum limit of line width is about
      \the\@tempcnta\@percentchar\MessageBreak
      larger than the heuristically detected line width.\MessageBreak
      You may e.g. increase DIV, decrease fontsize\MessageBreak
      or change the papersize%
    }
\fi\fi
```

```
    \ifnum\ta@div <6
      \PackageWarningNoLine{typearea}{%
        Very low DIV value!\MessageBreak
        DIV values less than 6 result in textwidth/-height\MessageBreak
        smaller than total marginwidth/-height.\MessageBreak
        You should e.g. increase DIV, increase fontsize or\MessageBreak
        change papersize%
      }
    \fi
    \ta@layoutinfo{\number\ta@div}{\the\@tempcnta}%
    \@typearea@end
    \activateareas
  \fi
}
```

`\ta@layoutinfo` Schreibt die Layout-Information in die `log`-Datei. Dabei ist das erste Argument die Ausgabe für den DIV-Wert und das zweite Argument die DIV-Abweichung. Letztere wird nur ausgegeben, wenn das Argument nicht leer und nicht `\relax` ist.

```
\newcommand*{\ta@layoutinfo}[2]{%
  \PackageInfo{typearea}{%
    These are the values describing the layout:\MessageBreak
    DIV\space\space=\space#1\MessageBreak
    BCOR\space=\space\the\ta@bcor\MessageBreak
    \string\paperwidth\space\space\space\space\space\space=%
                          \space\the\paperwidth\MessageBreak
    \space\string\textwidth\space\space\space\space\space\space=%
                          \space\the\textwidth\MessageBreak
    \ifx\relax#2\relax\else
      \space DIV departure\space\space\space=%
                            \space#2\@percentchar\MessageBreak
    \fi
    \space\string\evensidemargin\space=%
                          \space\the\evensidemargin\MessageBreak
    \space\string\oddsidemargin\space\space=%
                            \space\the\oddsidemargin\MessageBreak
    \string\paperheight\space\space\space\space\space=%
                          \space\the\paperheight\MessageBreak
    \space\string\textheight\space\space\space\space\space=%
                          \space\the\textheight\MessageBreak
    \space\string\topmargin\space\space\space\space\space\space=%
                          \space\the\topmargin\MessageBreak
    \space\string\headheight\space\space\space\space\space=%
                          \space\the\headheight\MessageBreak
    \space\string\headsep\space\space\space\space\space\space\space\space=%
                          \space\the\headsep\MessageBreak
    \space\string\topskip\space\space\space\space\space\space\space\space=%
                          \space\the\topskip\MessageBreak
    \space\string\footskip\space\space\space\space\space\space\space=%
```

```
                          \space\the\footskip\MessageBreak
    \space\string\baselineskip\space\space\space=%
                          \space\the\baselineskip\MessageBreak
  }%
}
```

\recalctypearea Dieses Anweisung dient der Neuberechnung des Satzspiegels mit den zuletzt eingestellten Werten. Dieser Befehl existiert erst, nachdem das Paket geladen wurde.

```
\AtEndOfPackage{\newcommand*{\recalctypearea}{\typearea[current]{last}}}
```

\areaset Zur Berechnung eines Satzspiegels mit fester Breite und Höhe dient der Befehl `\areaset`. Der Algorithmus ist ähnlich dem vorherigen Verfahren, aber einfacher. Auch hier soll ein Wechsel innerhalb des Dokuments ermöglicht werden.

```
\newcommand*\areaset[3][\z@]{%
  \edef\@tempa{#1}\edef\@tempb{current}\ifx\@tempa\@tempb\else
    \setlength{\ta@bcor}{#1}%
  \fi
```

Bevor nun irgendeine Berechnung durchgeführt wird, muss ggf. eine neue Seite begonnen werden.

```
  \if@atdocument\newpage\fi
```

Erst dann folgt die eigentlich Berechnung:

```
  \setlength{\textwidth}{#2}%
  \setlength{\textheight}{#3}%
  \def\@lastdiv{areaset}%
  \ta@hblk=\paperwidth
  \advance\ta@hblk by-\ta@bcor
  \advance\ta@hblk by-\textwidth
  \divide\ta@hblk by3
  \oddsidemargin=-1in
  \evensidemargin=-1in
  \if@twoside
    \if@semitwoside
      \advance\oddsidemargin by1.5\ta@hblk
      \advance\oddsidemargin by\ta@bcor
      \evensidemargin\oddsidemargin
    \else
      \advance\oddsidemargin by\ta@hblk
      \advance\oddsidemargin by\ta@bcor
      \advance\evensidemargin by2\ta@hblk
    \fi
  \else
    \advance\oddsidemargin by1.5\ta@hblk
    \advance\oddsidemargin by\ta@bcor
    \evensidemargin\oddsidemargin
  \fi
```

```
983   \scr@ifundefinedorrelax{@ptsize}{%
984     \topskip=\f@size\p@
985   }{%
986     \topskip=\@ptsize\p@
987     \advance\topskip by 10\p@
988   }%
989   \headheight=15\p@
990   \advance\headheight by\f@size\p@
991   \advance\headheight by-10\p@
992   \headsep=1.5\baselineskip
993   \footskip=3.5\baselineskip
994   \ta@vblk=\paperheight
995   \advance\ta@vblk by-\textheight
996   \divide\ta@vblk by3
997   \if@hincl
998     \advance\textheight by-\headheight
999     \advance\textheight by-\headsep
1000  \fi
1001  \if@fincl
1002    \advance\textheight by-\footskip
1003  \fi
1004  \topmargin \ta@vblk
1005  \advance\topmargin by-1in
1006  \if@hincl \else
1007    \advance\topmargin by-\headheight
1008    \advance\topmargin by-\headsep
1009  \fi
1010  \if@twoside
1011    \if@semitwoside
1012      \marginparwidth=\ta@hblk
1013    \else
1014      \marginparwidth=1.5\ta@hblk
1015    \fi
1016  \else
1017    \marginparwidth=\ta@hblk
1018  \fi
1019  \marginparsep=1cc
1020  \marginparpush=0.45\baselineskip
1021  \ta@layoutinfo{areaset}{}%
1022  \activateareas
1023 }
```

6.8 Berechnung des Standardsatzspiegels

Nun fehlt nur noch die eigentliche Initialisierung. Dazu wird einfach der Satzspiegel mit den getroffenen Voreinstellungen berechnet. Da der *DIV*-Wert 0 für einen Wert aus der Default-Liste steht, bleibt die Kompatibilität zu älteren Versionen erhalten. Der aktuelle Wert für *DIV* und *BCOR* wurde über die Optionen bereits gesetzt. Dabei kann abweichend zu sonst der

aktuelle Wert auch 0, 1, 2 oder 3 sein. Die Initialisierung des Pakets ist damit sehr einfach.

```
\typearea[current]{current}
```

Datei g
scrkfloa.dtx

7 Gleitumgebungen

Geplant!

KOMA-Script verfügt bezüglich der Gleitumgebungen über einige Features, die über das, was die Standardklassen bieten, hinaus geht. Allerdings ist das hier verwendete Interface gnadenlos veraltet und wird deshalb irgendwann ersetzt. Dabei wird höchstwahrscheinlich grundsätzlich auf das caption-Paket aufgebaut werden. Man betrachte dieses Modul daher als grundsätzlich zur Disposition stehend.

Die einzige Gleitumgebung, über die Briefe derzeit verfügen sind Randnotizen.

7.1 Einige Anweisungen und Einstellungen für die Optionen

Einige Anweisungen und Einstellungen werden bereits benötigt, bevor die Optionen abgearbeitet werden.

`\if@tablecaptionabove` `\@tablecaptionabovetrue` `\@tablecaptionabovefalse`

Die Einstellung, ob es Tabellenüber- oder -unterschriften sind, wird in einem Schalter gespeichert.

```
\newif\if@tablecaptionabove\@tablecaptionabovefalse
```

`\ifonelinecaptions` `\onelinecaptionstrue` `\onelinecaptionsfalse`

Die Entscheidung, ob die Sonderbehandlung für einzeilige caption-Texte durchgeführt werden soll, wird in einem Schalter gespeichert. In der Hoffnung, beim Laden von caption2 Speicher zu sparen, wird der gleiche Name verwendet, den auch caption2 verwendet. Trotzdem sollte man diesen Schalter als internen Schalter betrachten.

```
\newif\ifonelinecaptions\onelinecaptionstrue
```

`\scap@pos`

Hier wird gespeichert, ob die Beschreibung bei `captionbeside` links oder rechts vom Rest stehen soll. Voreingestellt ist rechts davon (***r**ight*). Andere Möglichkeiten sind ***l**eft*, ***o**uter*, ***i**nner*.

```
\newcommand*{\scap@pos}{r}
```

`\scap@valign`

Hier wird gespeichert, ob bei `captionbeside` die beiden Boxen (siehe Definition der Umgebung) ***t**op*, ***b**ottem* oder ***c**enter* ausgerichtet werden sollen. Andere Buchstaben sind nicht erlaubt. Voreingestellt ist ***b**ottem*.

```
\newcommand*{\scap@valign}{b}
```

7.2 Optionen für Titel von Gleitumgebungen

`captions` Dies ist nun die zentrale Option für Einstellungen zu `\caption`. Als besondere Schwierigkeit ergibt sich dabei jedoch, dass das Paket caption einige Sonderbehandlungen für KOMA-Script enthält, die darauf angewiesen sind, dass die alte Optionenschnittstelle verwendet wird. Erfreulicherweise funktioniert das aber ab Version 3.1g über eine Erweiterung der Schalter-Umschaltbefehle (`\@tablecaptionabovetrue`, `\@tablecaptionabovefalse`, `\onelinecaptionstrue`, `\onelinecaptionsfalse`), so dass lediglich dafür georgt werden muss, dass diese weiterhin verwendet werden.

```
\KOMA@key{captions}{%
  \KOMA@set@ncmdkey{captions}{@tempa}{%
    {tableheading}{0},%
    {tableabove}{0},{abovetable}{0},{abovetabular}{0},{topattable}{0},%
    {tablesignature}{1},%
    {tablebelow}{1},{belowtable}{1},{belowtabular}{1},{bottomattable}{1},%
    {oneline}{2},%
    {nooneline}{3},%
    {rightbeside}{4},{besideright}{4},%
    {leftbeside}{5},{besideleft}{5},%
    {innerbeside}{6},{besideinner}{6},%
    {outerbeside}{7},{besideouter}{7},%
    {topbeside}{8},{besidetop}{8},%
    {centeredbeside}{9},{besidecentered}{9},%
    {middlebeside}{9},{besidemiddle}{9},%
    {bottombeside}{10},{besidebottom}{10}%
  }{#1}%
  \ifcase \@tempa\relax
    \@tablecaptionabovetrue
  \or
    \@tablecaptionabovefalse
  \or
    \onelinecaptionstrue
  \or
    \onelinecaptionsfalse
  \or
    \renewcommand*{\scap@pos}{r}%
  \or
    \renewcommand*{\scap@pos}{l}%
  \or
    \renewcommand*{\scap@pos}{i}%
  \or
    \renewcommand*{\scap@pos}{o}%
  \or
    \renewcommand*{\scap@valign}{t}%
  \or
    \renewcommand*{\scap@valign}{c}%
  \or
```

```
    \renewcommand*{\scap@valign}{b}%
  \fi
}
```

tablecaptionabove
tablecaptionbelow

Es besteht nun die Wahl, ob `\caption` bei Tabellen für Unter- oder Überschriften voreingestellt ist. Die Defaulteinstellung ist aus Kompatibilitätsgründen die Tabellenunterschrift.

```
\KOMA@DeclareDeprecatedOption{tablecaptionabove}{captions=tableheading}
\KOMA@DeclareDeprecatedOption{tablecaptionbelow}{captions=tablesignature}
```

origlongtable

Wenn das longtable-Paket geladen ist, werden die Tabellentitel dafür normalerweise von den Klassen umdefiniert. Dies geschieht automatisch nicht, wenn das caption2-Paket geladen ist. Es geschieht ausserdem nicht, wenn diese Option gesetzt ist.

\if@komalongtable
\@komalongtabletrue
\@komalongtablefalse

Die Einstellung wird in einem Schalter gespeichert.

```
\newif\if@komalongtable\@komalongtabletrue
\DeclareOption{origlongtable}{\@komalongtablefalse}
```

onelinecaption
noonelinecaption

Da für diese Option nicht der gleiche Name verwendet wird, wie bei caption2, wurde von den Optionen bis Version 2.98b nicht nur der Schalter entsprechend gesetzt, sondern auch die korrespondierende Option an caption2 weitergeleitet. Der gleiche Name konnte schlecht verwendet werden, weil er bei einer Klasse auch nicht ansatzweise als selbsterklärend bezeichnet werden kann. Ab Version 2.98c entfällt die Sonderbehandlung von caption2.

```
\KOMA@DeclareDeprecatedOption{onelinecaption}{captions=oneline}
\KOMA@DeclareDeprecatedOption{noonelinecaption}{captions=nooneline}
```

7.3 Randnotizen

Randnotizen sind ebenfalls eine Art von Gleitumgebung.

\marginline

KOMA-Script stellt zusätzlich ein Randzeilenmakro zur Verfügung, das je nach gerader oder ungerader Seite in doppelseitigen Layout rechts- oder linksbündig gesetzt werden muss.

```
\newcommand\marginline[1]{%
  \marginpar[\raggedleft{#1}]{\raggedright #1}%
}
```

7.4 Definitionen für Titel von Gleitumgebungen

\caption

Es ist möglich, die Fließumgebungen mit einem Titel und einer Beschreibung zu versehen. Diese wird vom ggf. etwas abgesetzt.

abovecaptionskip
belowcaptionskip

Um wieviel ist in zwei Längenregistern gespeichert:

```
\newlength{\abovecaptionskip}
\newlength{\belowcaptionskip}
```

```
\setlength{\abovecaptionskip}{10\p@}
\setlength{\belowcaptionskip}{0\p@}
```

`\captionformat` Trennung von Label und Text:

```
\newcommand*{\captionformat}{:\ }
```

`\if@captionabove` Ob es sich um eine Unter- oder Überschrift handelt und also `\abovecaptionskip` davor und `\belowcaptionskip` danach steht oder umgekehrt, wird über diesen Schalter gesteuert. Voreingestellt sind Unterschriften.

```
\newif\if@captionabove\@captionabovefalse
```

`\scr@caption` `\captionabove` `\captionbelow` Damit überhaupt zwischen Unter- und Überschriften unterschieden werden kann, muss zunächst das Originalmakro gesichert werden. Dies geschieht erst zu Beginn des Dokuments, damit Pakete ggf. noch ihre eigenen Definitionen von `\caption` vornehmen können. Das hyperref-Paket ist so ein Kandidat.

```
\newcommand{\scr@caption}{}
\AtBeginDocument{\let\scr@caption\caption}
```

Jetzt ist es möglich durch einfaches Umlegen des Schalters für die korrekte Darstellung zu sorgen:

```
\newcommand{\captionabove}{\@captionabovetrue\scr@caption}
\newcommand{\captionbelow}{\@captionabovefalse\scr@caption}
```

`captionbeside` Um einiges komplizierter als die Beschreibung unter oder über die Tabelle oder Abbildung zu setzen, ist der Versuch, sie daneben zu setzen. Dies funktioniert frundsätzlich auch nicht mit `longtable`-Tabellen.

`\scap@caption` Die Beschreibung, die der Umgebung als Parameter übergeben wird, muss zwischengespeichert werden. Hier wird lediglich das Makro dafür reserviert.

```
\newcommand*{\scap@caption}{}
```

`\scap@width` Die Breite, die Tabelle bzw. Abbildung einschließlich Beschreibung und insgesamt haben dürften. Voreingestellt ist die gesamte verfügbare Breite.

```
\newcommand*{\scap@width}{\linewidth}
```

`\scap@move` Der Wert, um den die Tabelle bzw. Abbildung einschließlich Beschreibung nach rechts bzw. außen verschoben werden. Voreingestellt ist keine Verschiebung.

```
\newcommand*{\scap@move}{\z@}
```

`\scap@currentlabel` Da `\caption` innerhalb einer Umgebung (`\parbox` und `captionbeside`) aufgerufen wird, würde ein `\label`, das zwar innerhalb der Gleitumgebung aber nach der `captionbeside`-Umgebung eingefügt wird, nicht die Gleitumgebung, sondern die umgebende Referenzebene referenzieren. Damit das nicht geschieht, wird in der `captionbeside`-Umgebung

in der \parbox die aktuelle Referenz \@currentlabel in dem globalen Makro \scap@currentlabel zwischengespeichert und am Ende der captionbeside-Umgebung mit Hilfe von \aftergroup nach dem Ende der Umgebung wiederhergestellt. Da die captionbeside-Umgebung ohnehin nicht geschachtelt werden kann, funktioniert das problemlos. Für diese Aktion wird hier ein globales Makro reserviert.

```
\newcommand*{\scap@currentlabel}{}
```

\ifscap@alt
\scap@alttrue
\scap@altfalse

Der Schalter, der angibt, ob die Verschiebung \scap@move im doppelseitigen Modus aus linken Seiten vom rechten Rand aus nach links erfolgen soll.

```
\newif\ifscap@alt
```

\@captionbeside
\@@captionbeside
\@@@captionbeside
\@@@@captionbeside

Die Umgebung soll einmal folgende Parameter besitzen:

(optional) Verzeichniseintrag

(obligatorisch) Beschreibung

(optional) lokaler Wert für \scap@pos

(optional) lokaler Wert für \scap@width

(optional) lokaler Wert für \scap@move

(*) lokaler Wert für \ifscap@alt (Stern gesetzt entspricht \scap@alttrue

Wie man sieht, sollte man bei Angabe aller optionalen Parameter auch bewusst den Stern setzen oder darauf verzichten.

Damit alle Parameter ausgewertet werden können, werden einige Hilfsmakros benötigt.

```
\newcommand*{\@captionbeside}{}
\long\def\@captionbeside[#1]#2{%
  \@ifnextchar [%]
    {\@@captionbeside[{#1}]{#2}}
    {\c@pti@nb@sid@[{#1}]{#2}}%
}
\newcommand*{\@@captionbeside}{}
\long\def\@@captionbeside[#1]#2[#3]{%
  \def\scap@pos{#3}%
  \@ifnextchar [%]
    {\@@@captionbeside[{#1}]{#2}}
    {\c@pti@nb@sid@[{#1}]{#2}}%
}
\newcommand*{\@@@captionbeside}{}
\long\def\@@@captionbeside[#1]#2[#3]{%
  \def\scap@width{#3}%
  \@ifnextchar [%]
```

```
    {\@@@@captionbeside[{#1}]{#2}}
    {\setlength{\@tempdima}{#3}%
      \addtolength{\@tempdima}{-\linewidth}%
      \setlength{\@tempdima}{-.5\@tempdima}%
      \edef\scap@move{\the\@tempdima}%
      \c@pti@nb@sid@[{#1}]{#2}}%
}
\newcommand*{\@@@@captionbeside}{}
\long\def\@@@@captionbeside[#1]#2[#3]{%
  \def\scap@move{#3}%
  \@ifstar
    {\scap@alttrue\c@pti@nb@sid@[{#1}]{#2}}
    {\scap@altfalse\c@pti@nb@sid@[{#1}]{#2}}%
}
```

`\c@pti@nb@sid@b@x` Es wird noch eines der kostbaren Boxregister benötigt.

```
\newbox\c@pti@nb@sid@b@x
```

`\c@pti@nb@sid@` Hier wird nun die Umgebung eigentlich geöffnet, während zuvor nur die Parameter ausgewertet wurden.

```
\newcommand{\c@pti@nb@sid@}{}
\long\def\c@pti@nb@sid@[#1]#2{%
  \def\scap@caption{%
    \if c\scap@valign
      \setlength{\abovecaptionskip}{\z@}%
    \else \if t\scap@valign
        \setlength{\abovecaptionskip}{-\ht\strutbox}%
      \else \if b\scap@valign
          \setlength{\abovecaptionskip}{\z@}%
        \fi
      \fi
    \fi
    \setlength{\belowcaptionskip}{-\dp\strutbox}%
    \captionbelow[{#1}]{#2}%
  }%
  \begin{lrbox}{\c@pti@nb@sid@b@x}%
}
```

Jetzt endlich können wir uns auf die Umgebung stürzen. Die Umgebung zu starten ist einfach. Es wird ein Doppelargumentmakro ausgeführt.

```
\newenvironment*{captionbeside}{%
  \@dblarg\@captionbeside%
}{%
```

Komplizierter ist die Beendigung der Umgebung. Hier fällt der gesamte Satzaufwand an.

```
  \end{lrbox}%
  \settowidth{\@tempdima}{\usebox\c@pti@nb@sid@b@x}%
  \setlength{\@tempdimb}{\scap@width}%
```

```
  \addtolength{\@tempdimb}{-\@tempdima}%
  \addtolength{\@tempdimb}{-\columnsep}%
  \ifdim \@tempdimb<2em
    \ClassWarning{\KOMAClassName}{%
      caption beside will be very, very small.\MessageBreak
      In most cases this will result in overfull and\MessageBreak
      underfull \string\hbox\space messages%
    }%
  \fi
  \onelinecaptionsfalse
  \setcapwidth[r]{\@tempdimb}%
  \let\@tempa\scap@pos
  \setlength{\@tempdimc}{\scap@move}%
  \if i\scap@pos\def\@tempa{l}\fi
  \if o\scap@pos\def\@tempa{r}\fi
  \if@twoside
    \ifthispageodd{}{%
      \if o\scap@pos\def\@tempa{l}\fi
      \if i\scap@pos\def\@tempa{r}\fi
      \ifscap@alt
        \addtolength{\@tempdimc}{\scap@width}%
        \addtolength{\@tempdimc}{-\linewidth}%
        \setlength{\@tempdimc}{-\@tempdimc}%
      \fi
    }%
  \fi
  \hspace*{\@tempdimc}%
  \if l\@tempa
    \rlap{%
      \parbox[\scap@valign]{\@tempdimb}{\scap@caption
        \global\let\scap@currentlabel\@currentlabel}%
      \nobreak\hskip\columnsep
      \parbox[\scap@valign]{\wd\c@pti@nb@sid@b@x}{\usebox\c@pti@nb@sid@b@x}%
    }%
  \else
    \rlap{%
      \parbox[\scap@valign]{\wd\c@pti@nb@sid@b@x}{\usebox\c@pti@nb@sid@b@x}%
      \nobreak\hskip\columnsep
      \parbox[\scap@valign]{\@tempdimb}{\scap@caption
        \global\let\scap@currentlabel\@currentlabel}%
    }%
  \fi
  \par
  \aftergroup\let\aftergroup\@currentlabel
  \aftergroup\scap@currentlabel
}
```

`\capfont` Die Schriftart der Titel kann eingestellt werden. Dazu dient dieses Makro, das als internes Makro zu verstehen ist,

\caplabelfont Davon unabhängig kann auch die Schriftart der Titel-Titel („Abbildung X:“) eingestellt werden. Auch dieses Makro ist als internes Makro zu verstehen.

```
\newcommand*\capfont{\normalfont\normalcolor}
\newcommand*\caplabelfont{\normalfont\normalcolor}
```

\scr@fnt@caption
\scr@fnt@captionlabel Die eigentliche Änderung der Schriftart erfolgt über das Element `caption` bzw. `captionlabel`.

```
\newcommand*{\scr@fnt@caption}{\capfont}
\newcommand*{\scr@fnt@captionlabel}{\caplabelfont}
```

\@makecaption Bis Version 2.5f sah die Definition wie folgt aus.

```
\long\def\@makecaption#1#2{%
  \vskip\abovecaptionskip
  \sbox\@tempboxa{{\caplabelfont #1\captionformat}{\capfont #2}}%
  \ifdim \wd\@tempboxa >\hsize
    \@hangfrom{{\caplabelfont #1\captionformat}}{\capfont #2\par}
  \else
    \global \@minipagefalse
    \hb@xt@\hsize{\hfil\box\@tempboxa\hfil}%
  \fi
  \vskip\belowcaptionskip}
```

\ifdofullc@p Nun jedoch wird das ganze etwas komplizierter. Als erstes wird ein Schalter benötigt, der angibt, ob auf die volle Länge des „Labels“ eingezogen werden soll, oder nicht:

```
\newif\ifdofullc@p
```

wobei die Standardeinstellung der volle Einzug ist, um zum bisherigen Verhalten kompatibel zu bleiben.

```
\dofullc@ptrue
```

\if@capbreak Außerdem gibt es einen Schalter, der angibt, ob nach dem Label ein Umbruch erfolgen soll. Dies ist normalerweise nicht der Fall.

```
\newif\if@capbreak\@capbreakfalse
```

\setcapindent Dann wird weiter ein Befehl benötigt, mit dem der Benutzer die Weite des Einzugs selbst einstellen kann. Da intern für die Weite ein Makro verwendet werden soll, findet hier sicherheitshalber eine Zuweisung an eine Dimension statt. So sollten Fehlermeldungen hier und nicht erst bei Verwendung auftauchen. Natürlich muss hier auch der automatische Einzug abgeschaltet werden. In der Sternform wird außerdem dafür gesorgt, dass zwischen „Label“ und Beschreibung ein Umbruch erfolgt. In der normalen Version wird stattdessen für Kompatibilität zu früheren Versionen gesorgt.

\@setcapindent
\@@setcaptindent

Für die Implementierung der Sternform werden zwei interne Makros benötigt.

```
\newcommand*{\setcapindent}{\@ifstar{\@@setcapindent}{\@setcapindent}}
\newcommand*{\@setcapindent}[1]{%
  \setlength{\@tempdima}{#1}%
  \ifdim\@tempdima>\hsize
    \addtolength{\@tempdima}{-\hsize}\@capbreaktrue
  \else
    \@capbreakfalse
  \fi
  \edef\cap@indent{\the\@tempdima}%
  \dofullc@pfalse
}
\newcommand*{\@@setcapindent}[1]{%
  \setlength{\@tempdima}{#1}%
  \edef\cap@indent{\the\@tempdima}%
  \dofullc@pfalse\@capbreaktrue
}
\AfterPackage{caption2}{%
  \renewcommand*{\setcapindent}{%
    \ClassWarning{\KOMAClassName}{%
      You're using package `caption2'.\MessageBreak
      Because of this you should also use `\string\captionstyle'
      and\MessageBreak
      set length '\string\captionindent' instead of
      using\MessageBreak
      `\string\setcapindent'%
    }%
    \captionstyle{indent}%
    \@ifstar{\setlength{\captionindent}}{\setlength{\captionindent}}%
  }%
}
```

\setcaphanging

Andererseits soll es auch möglich sein, nachträglich wieder den automatischen Einzug einzuschalten. Dafür gibt es ein weiteres Makro.

```
\newcommand*{\setcaphanging}{\dofullc@ptrue\@capbreakfalse}
\AfterPackage{caption2}{%
  \renewcommand*{\setcaphanging}{%
    \ClassWarning{\KOMAClassName}{%
      You're using package `caption2'.\MessageBreak
      Because of this you should also use
      `\string\captionstyle'\MessageBreak
      instead of `\string\setcaphang'%
    }
    \captionstyle{hang}%
  }%
}
```

Nun aber zur neuen Definition der Titel. Zunächst geht alles wie gehabt.

```
\newcommand{\@makecaption}[2]{%
  \if@captionabove
    \vskip\belowcaptionskip
  \else
    \vskip\abovecaptionskip
  \fi
  \@@makecaption\@firstofone{#1}{#2}%
  \if@captionabove
    \vskip\abovecaptionskip
  \else
    \vskip\belowcaptionskip
  \fi
}
```

`\@@makecaption` Tatsächlich gesetzt wird der Titel mit diesem Makro. Dabei muss der erste Parameter entweder `\@gobble` oder `\@firstofone` sein.

`\cap@width` `\cap@left` `\cap@right` Die Breite der mehrzeiligen bzw. die maximale Breite der einzeiligen *captions* sowie das Makro das links und rechts von der Box mit der mehrzeiligen *caption* aufgerufen wird. In der Voreinstellung wird für Zentrierung gesorgt.

```
\newcommand*{\cap@width}{\hsize}
\newcommand*{\cap@left}{\hfill}
\newcommand*{\cap@right}{\hfill}
```

`\setcapwidth` Hier wird `\cap@width` gesetzt. Dabei werden optional auch `\cap@left` und `\cap@right` angepasst.

```
\newcommand*{\setcapwidth}[2][]{%
  \begingroup%
    \setlength{\@tempdima}{#2}%
    \ifdim\@tempdima <\z@
      \@tempdima\z@
      \ClassWarning{\KOMAClassName}{%
        caption width less than 0pt not allowed.\MessageBreak
        Setting caption width to 0pt%
      }%
    \fi
    \edef\@tempa{\endgroup\def\noexpand\cap@width{\the\@tempdima}}%
  \@tempa
  \begingroup
    \edef\@tempa{#1}%
    \def\@tempb{c}\ifx\@tempa\@tempb
      \endgroup
      \def\cap@margin{\let\cap@left=\hfill\let\cap@right=\hfill}%
    \else
      \def\@tempb{l}\ifx\@tempa\@tempb
        \endgroup
        \def\cap@margin{\let\cap@left=\relax\let\cap@right\hfill}%
```

```
      \else
        \def\@tempb{r}\ifx\@tempa\@tempb
          \endgroup
          \def\cap@margin{\let\cap@left=\hfill\let\cap@right=\relax}%
        \else
          \def\@tempb{i}\ifx\@tempa\@tempb
            \endgroup
            \def\cap@margin{%
              \let\cap@left=\relax
              \let\cap@right=\hfill
              \if@twoside
                \ifthispageodd{}{%
                  \let\cap@left=\hfill
                  \let\cap@right=\relax
                }%
              \fi
            }%
          \else
            \def\@tempb{o}\ifx\@tempa\@tempb
              \endgroup
              \def\cap@margin{%
                \let\cap@left=\hfill
                \let\cap@right=\relax
                \if@twoside
                  \ifthispageodd{}{%
                    \let\cap@left=\relax
                    \let\cap@right=\hfill
                  }%
                \fi
              }%
            \else
              \ifx \@tempa\@empty\else
                \ClassError{\KOMAClassName}{%
                  unknown optional position argument%
                }{%
                  You've used `#1' as position argument at
                  \string\setcaptionwidth,\MessageBreak
                  but only ONE of `c', `l', `r', `i`, `o` ist
                  allowed.\MessageBreak
                  Argument will be ignored, if you'll continue.%
                }%
              \fi
              \endgroup
  \fi\fi\fi\fi\fi
}
```

Falls caption2 geladen wird, ändert sich die Funktion. Das optionale Argument wird dann ignorriert.

```
\AfterPackage{caption2}{%
  \renewcommand*{\setcapwidth}[2][]{%
```

```
    \ClassWarning{\KOMAClassName}{%
      Ignorring optional argument of `\string\setcapwidth'!\MessageBreak
      You're using package `caption2'.\MessageBreak
      Because of this you should also use
      `\string\setcaptionwidth'\MessageBreak
      instead of `\string\setcapwidth'%
    }%
    \setcaptionwidth{#2}%
  }%
}%
```

`\cap@margin` Dieses Makro beinhaltet nicht den Wert der Ränder, sondern das Makro, mit dem innerhalb von `\@@makecaption` die Makros `\cap@left` und `\cap@right` neu definiert werden. Dies geschieht nur, wenn die Ränder mit `\setcapmargin` gesetzt werden. Vordefiniert ist also keinerlei Aktion.

```
\newcommand*{\cap@margin}{}
\let\cap@margin=\relax%
```

`\setcapmargin` `\@setcapmargin` `\@@setcapmargin` Statt die Breite der *caption* zu setzen, können auch die Ränder gesetzt werden. In der Sternvariante werden im zweiseitigen Modus die Ränder auf linken Seiten vertauscht.

```
\newcommand*{\setcapmargin}{\@ifstar{\@setcapmargin}{\@@setcapmargin}}
\newcommand*{\@setcapmargin}[2][\@tempdimb]{%
  \def\cap@margin{%
    \onelinecaptionsfalse
    \begingroup
      \setlength{\@tempdima}{\hsize}%
      \setlength{\@tempdimb}{#2}%
      \addtolength{\@tempdima}{-\@tempdimb}%
      \setlength{\@tempdimc}{#1}%
      \addtolength{\@tempdima}{-\@tempdimc}%
      \edef\@tempa{\endgroup
        \def\noexpand\cap@width{\the\@tempdima}%
        \noexpand\if@twoside
          \noexpand\ifthispageodd{%
            \def\noexpand\cap@left{\noexpand\hspace*{\the\@tempdimc}}%
            \def\noexpand\cap@right{\noexpand\hspace*{\the\@tempdimb}}%
          }{%
            \def\noexpand\cap@right{\noexpand\hspace*{\the\@tempdimc}}%
            \def\noexpand\cap@left{\noexpand\hspace*{\the\@tempdimb}}%
          }%
        \noexpand\else
          \def\noexpand\cap@left{\noexpand\hspace*{\the\@tempdimc}}%
          \def\noexpand\cap@right{\noexpand\hspace*{\the\@tempdimb}}%
        \noexpand\fi
      }%
    \@tempa
  }%
}
```

```
\newcommand*{\@@setcapmargin}[2][\@tempdimb]{%
  \def\cap@margin{%
    \onelinecaptionsfalse
    \begingroup
      \setlength{\@tempdima}{\hsize}%
      \setlength{\@tempdimb}{#2}%
      \addtolength{\@tempdima}{-\@tempdimb}%
      \setlength{\@tempdimc}{#1}%
      \addtolength{\@tempdima}{-\@tempdimc}%
      \edef\@tempa{\endgroup
        \def\noexpand\cap@width{\the\@tempdima}%
        \def\noexpand\cap@left{\noexpand\hspace*{\the\@tempdimc}}%
        \def\noexpand\cap@right{\noexpand\hspace*{\the\@tempdimb}}%
      }%
    \@tempa
  }%
}
```

Wenn caption2 geladen ist, ändert sich die Bedeutung.

```
\AfterPackage{caption2}{%
  \renewcommand*{\@setcapmargin}[2][\@tempdimb]{%
    \setlength{\@tempdimb}{#2}%
    \setlength{\@tempdima}{#1}%
    \ifdim\@tempdima\@tempdimb\else
      \ClassWarning{\KOMAClassName}{%
        Can set only centered caption!\MessageBreak
        You're using package `caption2'.\MessageBreak
        Because of this you should also use
        `\string\setcaptionmargin'\MessageBreak
        instead of `\string\setcapmargin'%
      }%
    \fi
    \addtolength{\@tempdimb}{\@tempdimb}%
    \setcaptionmargin{.5\@tempdimb}}%
  \let\@@setcapmargin=\@setcapmargin
}
```

Jetzt aber zum Eigentlichen. Die Breite der *caption* muss nur bestimmte werden, wenn Option `onelinecaption` verwendet wird. Das bedeutet auch, dass bei `noonelinecaption` das Problem mit dem Hochzählen der Fußnoten nicht auftritt.

```
\newcommand{\@@makecaption}[3]{%
  \cap@margin
  \ifonelinecaptions
    \@tempcnta=\value{mpfootnote}\@tempcntb=\value{footnote}%
    \let\scr@tmp@footnotetext=\@footnotetext
    \let\@footnotetext=\@gobble
    \sbox\@tempboxa{%
      {\capfont{\strut\ignorespaces
```

```
      #1{{\caplabelfont{#2\captionformat}}}%
      #3\unskip\strut}}%
  }%
  \let\@footnotetext=\scr@tmp@footnotetext
  \let\scr@tmp@footnotetext=\undefined
  \setcounter{footnote}{\@tempcntb}%
  \setcounter{mpfootnote}{\@tempcnta}%
\else
  \sbox\@tempboxa{\hspace*{2\cap@width}}%
\fi
\begingroup%
  \setlength{\@tempdima}{\cap@width}%
  \ifdim \wd\@tempboxa >\@tempdima
    \hb@xt@\hsize{%
      \setbox\@tempboxa\vbox{\hsize=\cap@width
```

Im Fall, dass nun also mehrzeilig gesetzt werden muss, ist zu unterscheiden, ob wie bisher voll hängend gearbeitet werden soll oder nicht. In ersterem Fall machen wir das, wie bisher, allerdings wird der `\capfont` früher aktiviert, damit darin auch globale Größsenänderungen für den Titel Platz finden. Außerdem wird die geänderte Parameterübergabe berücksichtigt.

```
        \ifdofullc@p
          {\capfont{\@hangfrom{\scr@smashdp{\strut\ignorespaces
                #1{{\caplabelfont{#2\captionformat}}}}}%
            {#3\unskip\strut\par}}}%
```

Im zweiten Fall ist nun zu unterscheiden, ob nach dem Label ein Umbruch erfolgen soll oder nicht. Dabei ist der Einzug zu beachten. Ist der Einzug negativ, so findet ebenfalls ein Umbruch statt und nur die erste Folgezeile wird eingezogen.

```
        \else%
          \ifdim\cap@indent<\z@
            {\capfont{\strut\ignorespaces
              #1{{\caplabelfont{#2\captionformat\par}}}%
              \noindent\hspace*{-\cap@indent}#3\unskip\strut\par}}%
          \else%
            \if@capbreak
              {\capfont{\strut\ignorespaces
                #1{{\caplabelfont{#2\captionformat\par}}}%
                \noindent\@hangfrom{\hspace*{\cap@indent}}{#3\par}}}%
            \else
              {\capfont{\@hangfrom{\scr@smashdp\strut\hspace*{\cap@indent}}%
                {\hspace{-\cap@indent}\scr@smashdp{\ignorespaces#1%
                  {{\caplabelfont{#2\captionformat}}}}#3\unskip\strut\par}}}%
            \fi
          \fi
        \fi
      }%
      \setlength{\@tempdima}{\ht\@tempboxa}%
```

```
        \addtolength{\@tempdima}{\dp\@tempboxa}%
        \addtolength{\@tempdima}{-\ht\strutbox}%
        \ht\@tempboxa\ht\strutbox
        \dp\@tempboxa\@tempdima
        \strut\cap@left
        \box\@tempboxa
        \cap@right\strut
      }%
    \else
      \global \@minipagefalse
```

Im einzeiligen Fall wird nun die Box wie oben erneut erzeugt und gleich gesetzt. Im Prinzip ginge das auch direkt ohne erst eine Box zu erzeugen.

```
      \sbox\@tempboxa{%
        {\capfont{\scr@smashdp{\strut\ignorespaces
          #1{{\caplabelfont{#2\captionformat}}}}%
          #3}}%
      }%
      {\capfont{\hb@xt@\hsize{\strut\cap@left\box\@tempboxa\cap@right\strut}}}%
    \fi
  \endgroup
}
```

\LT@makecaption
\LT@array
\LT@caption
tion@before@longtable@warning

Ab Version 2.8j wird auf vielfachen Wunsch das longtable-Paket berücksichtigt. Da Pakete aber nach Klassen geladen werden, kann dies erst nach dem Laden des Pakets aktiviert werden. Die Definition entspricht der obigen. Wird das caption2-Paket nach longtable geladen, so wird automatisch die Definition von caption2 verwendet. Wird caption2 vor longtable geladen, so liegt voraussichtlich eine Fehlbedienung vor, auf die hingewiesen wird.

```
\newcommand*{\scr@caption@before@longtable@warning}[1]{%
  \ClassWarning{\KOMAClassName}{%
    you've loaded package `#1' before package\MessageBreak
    `longtable'!\MessageBreak
    Because of this package `#1' isn't able to\MessageBreak
    control `\string\caption' at longtables. \space After loading
    \MessageBreak
    package `#1' KOMA-Script also doesn't control\MessageBreak
    `\string\caption'.\MessageBreak
    I think, you should load package `caption' after\MessageBreak
    package `longtable'%
  }%
}
\AfterPackage{longtable}{%
  \@ifpackageloaded{caption2}{\@komalongtablefalse
    \scr@caption@before@longtable@warning{caption2}%
  }{}%
  \if@komalongtable
    \renewcommand{\LT@makecaption}[3]{%
      \noalign{%
```

```
        \if@captionabove
          \vskip\belowcaptionskip
        \else
          \vskip\abovecaptionskip
        \fi
      }%
      \LT@mcol\LT@cols c{%
        \hbox to\z@{\hss\parbox[t]\linewidth{%
            \@@makecaption{#1}{#2}{#3}%
            \endgraf
            \if@captionabove
              \vskip\abovecaptionskip
            \else
              \vskip\belowcaptionskip
            \fi
          }%
          \hss
        }%
      }%
    }%
    \newcommand*{\scr@LT@array}{}%
    \let\scr@LT@array=\LT@array
    \renewcommand*{\LT@array}{%
      \let\scr@caption=\LT@caption
      \def\captionabove{\noalign{\global\@captionabovetrue}%
        \scr@caption}%
      \def\captionbelow{\noalign{\global\@captionabovefalse}%
        \scr@caption}%
      \if@tablecaptionabove
        \let\LT@caption=\captionabove
      \else
        \let\LT@caption=\captionbelow
      \fi
      \scr@LT@array
    }%
    \ClassInfo{\KOMAClassName}{%
      longtable captions redefined%
    }%
  \fi
}
```

`\fs@komaabove`
`\fs@komabelow`
`\floatc@komascript`

Ab Version 2.8j werden für das float-Paket zwei float-Stile Namens `komaabove` und `komabelow` definiert. Da möglicherweise auch andere Pakete darauf zurückgreifen, wird das unabhängig von der Existenz von float getan. Verwendet wird dies innerhalb von KOMA-Script jedoch nicht. Es werden außerdem interne Makros von float verwendet!

```
\newcommand*{\fs@komaabove}{%
  \let\@fs@capt\floatc@komascript
  \let\@fs@pre\@empty
```

```
  \def\@fs@mid{\vskip\abovecaptionskip}%
  \let\@fs@post\@empty
  \let\@fs@iftopcapt\iftrue}
\newcommand*{\fs@komabelow}{%
  \let\@fs@capt\floatc@komascript
  \let\@fs@pre\@empty
  \def\@fs@mid{\vskip\abovecaptionskip}%
  \let\@fs@post\@empty
  \let\@fs@iftopcapt\iffalse}
\newcommand{\floatc@komascript}{\@@makecaption\@firstofone}
```

7.5 Verteilung der Gleitumgebungen

topnumber
\topfraction
bottomnumber
\bottomfraction
totalnumber
\textfraction
\floatpagefraction
dbltopnumber
\dbltopfraction
\dblfloatpagefraction

Für die Fließumgebungen müssen verschiedene Konstanten definiert werden, die Aufteilung zwischen Text und Fließumgebung und Ähnliches regeln.

```
\setcounter{topnumber}{2}
\renewcommand*\topfraction{.7}
\setcounter{bottomnumber}{1}
\renewcommand*\bottomfraction{.3}
\setcounter{totalnumber}{3}
\renewcommand*\textfraction{.2}
\renewcommand*\floatpagefraction{.5}
\setcounter{dbltopnumber}{2}
\renewcommand*\dbltopfraction{.7}
\renewcommand*\dblfloatpagefraction{.5}
```

7.6 Titel ohne Gleitumgebung

captionof

Obwohl es inzwischen mehrere Pakete gibt, die diese Anweisung bereit stellen, kommt es doch immer wieder zu Fehlern, weil einige Anwender diese Pakete nicht nutzen. Also habe ich beschlossen, das selbst zu implementieren. Um die Kompatibilität mit capt-of nicht zu stören, wird die Definition vor dme Laden jenes Pakets zurück genommen. Mit dem Paket caption gibt es solche Probleme nicht, das dieses `\captionof` ohnehin per `\def` definiert.

```
\newcommand*{\captionof}[1]{\def\@captype{#1}\caption}
\BeforePackage{capt-of}{\let\captionof\relax}
```

7.7 Standardbezeichner

\figurename
\tablename

So heißen die Dinger:

```
\newcommand*\figurename{Figure}
\newcommand*\tablename{Table}
```

7.8 Abbildungsumgebung

figure Zunächst werden figures, also Abbildungen behandelt.

figure
\thefigure
\figureformat

Dafür wird ein neuer Zähler benötigt, der bei der Buch- und der Bericht-Klasse mit \chapter zurückgesetzt wird und auch den Kapitelzähler mit enthält.

```
⟨book | report⟩\newcounter{figure}[chapter]
⟨article⟩\newcounter{figure}
\renewcommand*\thefigure{%
⟨*book⟩
  \ifnum \scr@compatibility>\@nameuse{scr@v@2.97d}\relax
    \if@mainmatter\thechapter.\fi
  \else
⟨/book⟩
⟨book | report⟩  \thechapter.%
⟨book⟩  \fi
  \@arabic\c@figure
}
\newcommand*{\figureformat}{%
  \figurename~\thefigure%
⟨report | book⟩  \autodot%
}
```

\fps@figure Die Standardplatzierung für Abbildungen ist *t*op, *b*ottom, *p*age.

```
\newcommand*{\fps@figure}{tbp}
```

\ftype@figure Um die verschiedenen *float*-Typen unterscheiden zu können, benätigen sie eine Nummer. Abbildungen bekommen hier willkürlich die Nummer 1.

```
\newcommand*{\ftype@figure}{1}
```

\ext@figure Die Extension der Datei für das Abbildungsverzeichnis.

```
\newcommand*{\ext@figure}{lof}
```

\fnum@figure Die Ausgabe der Abbildungsnummer.

```
\newcommand*{\fnum@figure}{\figureformat}
```

Jetzt aber zurück zur Umgebung. Wie üblich gibt es eine Version mit und eine Version ihne Stern. Zur unterschiedlichen Bedeutung bei zweispaltigem Satz siehe classes.dtx und source2e.

```
\newenvironment{figure}{%
  \let\caption\captionbelow\@float{figure}%
}{%
  \end@float
}
\newenvironment{figure*}{%
  \let\caption\captionbelow\@dblfloat{figure}%
}{%
  \end@dblfloat
}
```

7.9 Tabellengleitumgebung

table Auch verschiebbare Tabellen sind möglich. Das Prinzip ist das gleiche wie bei den Abbildungen.

table
\thetable
\tableformat

```
⟨article⟩\newcounter{table}
⟨book | report⟩\newcounter{table}[chapter]
\renewcommand*\thetable{%
⟨*book⟩
  \ifnum \scr@compatibility>\@nameuse{scr@v@2.97d}\relax
    \if@mainmatter\thechapter.\fi
  \else
⟨/book⟩
⟨book | report⟩  \thechapter.%
⟨book⟩  \fi
  \@arabic\c@table
}
\newcommand*{\tableformat}{%
  \tablename~\thetable%
⟨report | book⟩  \autodot%
}
```

\fps@table

```
\newcommand*{\fps@table}{tbp}
```

\ftype@table Allerdings wird für Tabellen willkürlich Nummer 2 verwendet.

```
\newcommand*{\ftype@table}{2}
```

\ext@table Natürlich haben sie auch eine andere Extension.

```
\newcommand*{\ext@table}{lot}
```

\fnum@table Die Ausgabe der Nummer kann ebenfalls abweichen.

```
\newcommand*{\fnum@table}{\tableformat}
```

Ob es \caption in der Voreinstellung eine Tabellenüberschrift oder eine Tabellenunterschrift erzeugt, hängt hier von den Optionen ab.

```
\newenvironment{table}{%
  \if@tablecaptionabove\let\caption\captionabove
  \else\let\caption\captionbelow\fi
  \@float{table}%
}{%
  \end@float
}
\newenvironment{table*}{%
  \if@tablecaptionabove\let\caption\captionabove
  \else\let\caption\captionbelow\fi
  \@dblfloat{table}%
}{%
```

```
  \end@dblfloat
}
```

Datei h
scrkftn.dtx

8 Fußnoten

Die Fußnotengestaltung in KOMA-Script basiert auf einem Hinweis aus dem dokumentierten LaTeX-Kern.

8.1 Option

`footnotes` Mehrfachfußnoten gibt es dann, wenn zu einer Textstelle mehrere Fußnoten hintereinander gesetzt werden. Normalerweise werden die einfach ohne Abstand hintereinander geklatscht. Man kann dann aber die beiden Fußnoten 1 und 2 beispielsweise nicht von der Fußnote 12 unterscheiden. Besser ist es, wenn mehrere Fußnoten automatisch durch ein Trennzeichen getrennt werden. Die Funktion ist footmisc entnommen und sollte inklusive Ein- und Ausschalten auch zusammen mit diesem Paket funktionieren.

```
\KOMA@key{footnotes}{%
  \ifx\@footnotemark\scr@saved@footnotemark
  \else
    \ifx\@footnotemark\scr@footnotemark
    \else
⟨class⟩      \ClassWarning{\KOMAClassName}{%
⟨package⟩      \PackageWarning{scrextend}{%
        Change of ‘\string\@footnotemark’ detected!\MessageBreak
        Use of ‘footnotes=#1’ may break\MessageBreak
        another package!\MessageBreak
        Maybe you should remove the usage of\MessageBreak
        option ‘footnotes=#1’}%
    \fi
  \fi
  \ifstr{#1}{multiple}{%
    \let\@footnotemark\scr@footnotemark
    \let\FN@mf@prepare\scr@mf@prepare
    \def\scr@footmisc@options{multiple}%
  }{%
    \ifstr{#1}{nomultiple}{%
      \let\@footnotemark\scr@saved@footnotemark
      \let\FN@mf@prepare\relax
      \let\scr@footmisc@options\@empty
    }{%
      \KOMA@unknown@keyval{footnotes}{#1}{‘multiple’ and ‘nomultiple’}%
    }%
  }%
}
```

\scr@footmisc@options

Für etwas mehr Kompatibilität mit footmisc, wird die aktuelle Einstellung der `multifootnotes` an das Paket weitergereicht.

```
\newcommand*{\scr@footmisc@options}{}
\PassOptionsToPackage{\noexpand\scr@footmisc@options}{footmisc}
```

\scr@saved@footnotemark
\scr@footnotemark

```
\newcommand*{\scr@saved@footnotemark}{%
  \leavevmode
  \ifhmode\edef\@x@sf{\the\spacefactor}\nobreak\fi
  \@makefnmark
  \ifhmode\spacefactor\@x@sf\fi
  \relax}
\expandafter\CheckCommand\expandafter*\expandafter\@footnotemark
\expandafter{\scr@saved@footnotemark}
\newcommand*{\scr@footnotemark}{%
  \leavevmode
  \ifhmode\edef\@x@sf{\the\spacefactor}\FN@mf@check\nobreak\fi
  \@makefnmark
  \csname FN@mf@prepare\endcsname
  \ifhmode\spacefactor\@x@sf\fi
  \relax}
```

8.2 Definitionen für Fußnoten

\footnoterule

Im Fall, dass `\raggedbottom` verwendet wird, die Fußnotenlinie nach unten gedrückt. Dies funktioniert nur dann nicht, wenn Abbildungen oder Tabellen mit der Option „`b`“ an das Ende der Seite gesetzt werden. `\raggedbottom` wird daran erkannt, dass `\@textbottom` nicht `\relax` ist. Das kann natürlich auch einmal schief gehen, ist dann aber auch nicht ganz so tragisch.

```
⟨*class⟩
\renewcommand*\footnoterule{%
  \normalsize\ftn@rule@test@values
  \kern-\dimexpr 2.6\p@+\ftn@rule@height\relax
  \ifx\@textbottom\relax\else\vskip \z@ \@plus.05fil\fi
  {\usekomafont{footnoterule}{%
      \hrule \@height\ftn@rule@height \@width\ftn@rule@width}}%
  \kern 2.6\p@}
⟨/class⟩
```

\ftn@rule@test@values

Test, ob die Einstellungen für die Längen der Fußnotenlinie halbwegs vernünftig sind.

```
\newcommand*{\ftn@rule@test@values}{%
  \ifdim\ftn@rule@height <\z@
    \ClassWarning{\KOMAClassName}{%
      You might get into trouble, because the\MessageBreak
      height of the footnote rule has a value\MessageBreak
      less than 0%
```

```
    }%
  \else
    \ifdim\ftn@rule@height >\dimexpr\skip\footins -2.6\p@\relax
      \ClassWarning{\KOMAClassName}{%
        You might get into trouble, because the\MessageBreak
        height of the footnote rule has a value\MessageBreak
        greater than \the\dimexpr\skip\footins -2.6\p@\relax
      }%
    \fi
  \fi
  \ifdim\ftn@rule@width <\z@
    \ClassWarning{\KOMAClassName}{%
      You might get into trouble, because the width\MessageBreak
      or length of the footnote rule has a value\MessageBreak
      less than 0pt%
    }%
  \else
    \ifdim\ftn@rule@width >\columnwidth
      \ClassWarning{\KOMAClassName}{%
        You might get into trouble, because the width\MessageBreak
        or length of the footnote rule has a value\MessageBreak
        greater than \string\columnwidth
      }%
    \fi
  \fi
}
```

`footnoterule` Damit man die Farbe der Fußnotentrennlinie ändern kann, gibt es nun ein eigenes Element dafür. Die Voreinstellung ist jedoch leer.

```
\newkomafont{footnoterule}{}
```

`\ftn@rule@width` Die Länge der Fußnotentrennlinie.

```
\newcommand*{\ftn@rule@width}{.4\columnwidth}
```

`\ftn@rule@height` Die Höhe der Fußnotentrennlinie (die Tiefe ist immer 0).

```
\newcommand*{\ftn@rule@height}{.4\p@}
```

`\setfootnoterule` Das erste, optionale Argument ist die Höhe der Linie, das zweite nicht optionale die Länge.

```
\newcommand*{\setfootnoterule}[2][]{%
  \ifstr{#1}{}{}{%
    \renewcommand*{\ftn@rule@height}{#1}%
  }%
  \ifstr{#2}{}{}{%
    \renewcommand*{\ftn@rule@width}{#2}%
  }%
  \ftn@rule@test@values
}
```

`footnote` Der Fußnotenzähler wird mit `\chapter` zurückgesetzt, die Fußnoten werden also kapitelweise nummeriert.

```
⟨book | report⟩\@addtoreset{footnote}{chapter}
```

`\deffootnote` Dieses Makro zur Definition der Gestalt von Fußnoten erlaubt einen optionalen und erwartet drei weitere Parameter. Der erste, optionale gibt den Einzug der ersten Zeile des Fußnotentextes vom linken Rand an. Die Fußnotenmarkierungen werden rechtsbündig in diesen Einzug gesetzt. Der zweite, also erste nicht optionale Parameter gibt den Einzug jeder weiteren Zeile des Fußnotentextes vom linken Rand an. Fehlt der optionale Parameter so ist er gleich diesem. Der dritte, also zweite nicht optionale Parameter gibt den zusätzlichen Einzug jedes weiteren Absatzes einer Fußnote an. Der letzte Parameter schließlich bestimmt die Ausgabe der Fußnotenmarkierung in der Fußnote. Diese wird zusätzlich in eine `\hbox` gesetzt.

```
\newcommand\deffootnote[4][]{%
  \long\def\@makefntext##1{%
    \setlength{\@tempdimc}{#3}%
    \def\@tempa{#1}\ifx\@tempa\@empty
      \@setpar{\@@par
        \@tempdima = \hsize
        \addtolength{\@tempdima}{-#2}%
        \parshape \@ne #2 \@tempdima}%
    \else
      \addtolength{\@tempdimc}{#2}%
      \addtolength{\@tempdimc}{-#1}%
      \@setpar{\@@par
        \@tempdima = \hsize
        \addtolength{\@tempdima}{-#1}%
        \@tempdimb = \hsize
        \addtolength{\@tempdimb}{-#2}%
        \parshape \tw@ #1 \@tempdima #2 \@tempdimb
      }%
    \fi
    \par
    \parindent\@tempdimc\noindent
    \ftn@font\hbox to \z@{\hss\@@makefnmark}##1%
  }%
```

`\@@makefnmark` Makro zum Setzen der Fußnotenmarkierung in der Fußnote:

```
  \def\@@makefnmark{\hbox{\ftnm@font{#4}}}%

}
```

`\ftn@font`
`\scr@fnt@footnote`
`\ftnm@font`
`\scr@fnt@footnotenumber`
`\scr@fnt@footnotenlabel`
`\scr@fnt@instead@footnotetext`

Das erste Element gibt die Schrift an, in der die Fußnote gesetzt wird. Davon abweichend kann mit dem zweiten Element die Schriftart für die Fußnotennummer getrennt angegeben werden. Aufgrund der Definition kann auch ein Makro verwendet werden, das ein Argument erwartet.

```
\newcommand*{\ftn@font}{\normalfont}
\newcommand*{\scr@fnt@footnote}{\ftn@font}
\newcommand*{\ftnm@font}{}
\newcommand*{\scr@fnt@footnotenumber}{\ftnm@font}
\newcommand*{\scr@fnt@footnotelabel}{\ftnm@font}
\newcommand*{\scr@fnt@instead@footnotetext}{footnote}
```

\deffootnotemark

Makro zur Definition der Fußnotenmarkierung im Text:

```
\newcommand*\deffootnotemark[1]{%
  \def\@makefnmark{\hbox{\ftntm@font{#1}}}%
}
```

\ftntm@font
\scr@fnt@footnotereference
\scr@fnt@footnoteref

Dies ist die Schriftart der Fußnotenreferenz im Text. Auch hier kann ggf. ein Makro verwendet werden, das ein Argument erwartet.

```
\newcommand*{\ftntm@font}{}
\newcommand*{\scr@fnt@footnotereference}{\ftntm@font}
\newcommand*{\scr@fnt@footnoteref}{\ftntm@font}
```

\FN@mf@check

Dafür sorgen, dass hier ggf. der Trenner eingefügt wird.

```
⟨package⟩\providecommand*{\FN@mf@check}{%
⟨class⟩\newcommand*{\FN@mf@check}{%
  \ifdim\lastkern=\multiplefootnotemarker\relax
    \edef\@x@sf{\the\spacefactor}%
    \unkern\multiplefootnoteseparator
    \spacefactor\@x@sf\relax
  \fi
}
```

\scr@mf@prepare

Damit \FN@mf@check informiert ist, dass es etwas zu tun hat.

```
\newcommand*{\scr@mf@prepare}{%
  \kern-\multiplefootnotemarker
  \kern\multiplefootnotemarker\relax
}
% \end{macro}
% \begin{macro}{\F@mf@prepare}
%   \changes{v2.98c}{2008/02/14}{Neu (intern)}
% Aus Gründen der Kompatibilität mit \textsf{footmisc} wird das dann
% vorsoglich auch noch definiert (und zwar als \cs{relax}, wenn es bisher noch
% nicht definiert ist).
\@ifundefined{FN@mf@prepare}{}{}
```

\@footnotetext
\scr@saved@footnotetext

Auch hier muss dafür gesorgt werden, dass die Markierung erfolgt.

```
\newcommand{\scr@saved@footnotetext}{}
\let\scr@saved@footnotetext\@footnotetext
\renewcommand{\@footnotetext}[1]{%
  \scr@saved@footnotetext{#1}%
  \csname FN@mf@prepare\endcsname
}
```

Es muss auch dafür gesorgt werden, dass footmisc nicht unnötig mit Warnung um sich wirft.

```
\BeforePackage{footmisc}{%
  \ifx\@footnotemark\scr@footnotemark
    \let\@footnotemark\scr@saved@footnotemark
  \fi
  \let\@footnotetext\scr@saved@footnotetext
}
```

\multiplefootnoteseparator Das ist der formatierte Separator!

```
\newcommand*{\multiplefootnoteseparator}{%
  \begingroup\let\thefootnotemark\multfootsep\@makefnmark\endgroup
}
```

\multfootsep Und hier footmisc-kompatibel unformatiert

```
⟨class⟩\newcommand*{\multfootsep}{,}
⟨package⟩\providecommand*{\multfootsep}{,}
```

\multiplefootnotemarker Das Markierungskerning wieder footmisc-kompatibel.

```
\newcommand*{\multiplefootnotemarker}{3sp}
```

\thefootnotemark Makro, damit \@thefnmark auf Anwenderebene verfügbar wird:

```
⟨class⟩\newcommand*{\thefootnotemark}{\@thefnmark}
⟨package⟩\providecommand*{\thefootnotemark}{\@thefnmark}
```

\@makefnmark

```
⟨*class⟩
\deffootnote[1em]{1.5em}{1em}{\textsuperscript{\thefootnotemark}}
\deffootnotemark{\textsuperscript{\thefootnotemark}}
⟨/class⟩
```

8.3 Fußnotenreferenz

\footref Damit man nicht nur zu diesem Zweck das Paket footmisc laden muss, kann man jetzt auch mit KOMA-Script alleine Fußnoten setzen, die Referenzen auf andere Fußnoten sind. Der Code stammt einerseits aus footmisc andererseits auf der Usenet-Gruppe comp.text.tex. Wer den zuerst hatte, ist für mich nicht mehr reproduzierbar. Jedenfalls steht er in footmisc unter LPPL und KOMA-Script steht ebenfalls unter LPPL. Das sollte also im Zweifelsfall nicht das Problem sein.

```
⟨class⟩\newcommand*{\footref}[1]{%
⟨package⟩\providecommand*{\footref}[1]{%
  \begingroup
    \unrestored@protected@xdef\@thefnmark{\ref{#1}}%
  \endgroup
  \@footnotemark
}
```

Datei i scrkpage.dtx

9 Seitenstil

Geplant!

Neben den Definitionen zum Seitenstil in dieser Datei gibt es auch noch das Paket scrpage2 (siehe `scrpage.dtx`). Es ist deshalb davon auszugehen, dass zukünftig die Definitionen hier komplett durch scrpage2 ersetzt werden und scrpage2 von den KOMA-Script-Klassen direkt geladen wird.

9.1 Optionen für die Trennlinien in Kopf und Fuß

`headsepline`
`headnosepline`
`\if@hsl`
`\@hsltrue`
`\@hslfalse`

Der Kopfteil kann mit einer Linie vom Textbereich getrennt werden.

Die Entscheidung wird in einem Schalter gespeichert.

```
\newif\if@hsl
```

Gleichzeitig wird dem typearea-Paket mitgeteilt, dass der Kopf zum Textbereich zu rechnen ist. Dies geschieht allerdings in umgekehrter Richtung nicht. Die Optionen sind also nicht symmetrisch.

```
\KOMA@key{headsepline}[true]{%
  \KOMA@set@ifkey{headsepline}{@hsl}{#1}%
  \KOMA@options{headinclude=#1}{}%
}
\KOMA@DeclareDeprecatedOption{headnosepline}{headsepline=false}
```

`footsepline`
`footnosepline`
`\if@fsl`
`\@fsltrue`
`\@fslfalse`

Der Fußteil kann mit einer Linie vom Textbereich getrennt werden.

Die Entscheidung wird in einem Schalter gespeichert.

```
\newif\if@fsl
```

Gleichzeitig wird dem `typearea`-Paket mitgeteilt, dass der Fuß zum Textbereich zu rechnen ist. Auch hier sind die Optionen nicht symmetrisch.

```
\KOMA@key{footsepline}[true]{%
  \KOMA@set@ifkey{footsepline}{@fsl}{#1}%
  \KOMA@options{footinclude=#1}{}%
}
\KOMA@DeclareDeprecatedOption{footnosepline}{footsepline=false}
```

In früheren Versionen wurde noch die Option `mpinclude` von den Klassen ausgewertet. Dies geschieht jetzt nur noch in typearea. Von den Klassen wird dann ggf. auf den dort definierten Schalter zugegriffen. Dieses Vorgehen ist deshalb sinnvoll, weil die Klassen ausnahmslos das typearea-Paket laden.

9.2 Optionen und Anweisungen für leere Seiten

\if@twoside

Die Voreinstellung wird direkt passend gesetzt.

```
⟨article | report | letter⟩\@twosidefalse
⟨book⟩\@twosidetrue
```

cleardoublepage
cleardoubleempty
cleardoubleplain
cleardoublestandard
\cleardoublestandardpage
\cleardoubleusingstyle
\cleardoubleemptypage
\cleardoubleplainpage
\cleardoubleoddpage
\cleardoubleoddstandardpage
\cleardoubleoddusingstyle
\cleardoubleoddemptypage
\cleardoubleoddplainpage
\cleardoubleevenpage
\cleardoubleevenstandardpage
\cleardoubleevenusingstyle
\cleardoubleevenemptypage
\cleardoubleevenplainpage
\cleardoublepage

Bei Verwendung von `twoside` und `openright` bei den Klassen scrbook und scrreprt wird bei `\chapter` implizit `\cleardoublepage` ausgeführt. Das führt ggf. normalerweise zu einer linken Seite, auf der noch der Kolumnentitel des vorherigen Kapitels und eine Seitenzahl steht. Häufig wird stattdessen nur eine Seitenzahl oder auch gar nichts gewünscht. Es soll also der Seitenstil `plain` oder `empty` verwendet werden. Dies wird mit den Optionen ermöglicht. Dazu werden auch gleich neue Makros definiert:

```
\KOMA@key{cleardoublepage}{%
  \begingroup%
    \def\@tempc{%
      \endgroup%
      \KOMA@unknown@keyval{cleardoublepage}{#1}{%
        'current' or any defined pagestyle e.g. 'empty','plain'}%
    }%
    \ifstr{#1}{current}{%
      \def\@tempc{\endgroup%
        \def\cleardoublepage{\cleardoublestandardpage}%
        \def\cleardoubleoddpage{\cleardoubleoddstandardpage}%
        \def\cleardoubleevenpage{\cleardoubleevenstandardpage}%
      }%
    }{%
      \ifnotundefined{ps@#1}{%
        \def\@tempc{\endgroup%
          \def\cleardoublepage{\cleardoublepageusingstyle{#1}}%
          \def\cleardoubleoddpage{\cleardoubleoddpageusingstyle{#1}}%
          \def\cleardoubleevenpage{\cleardoubleevenpageusingstyle{#1}}%
        }%
      }{}%
    }%
  \@tempc
}
\newcommand*{\cleardoubleoddstandardpage}{}
\let\cleardoubleoddstandardpage=\cleardoublepage
\newcommand*{\cleardoubleoddpageusingstyle}[1]{\clearpage
  {\pagestyle{#1}\cleardoubleoddstandardpage}}
\newcommand*{\cleardoubleoddemptypage}{\cleardoubleoddpageusingstyle{empty}}
\newcommand*{\cleardoubleoddplainpage}{\cleardoubleoddpageusingstyle{plain}}
\newcommand*{\cleardoubleevenstandardpage}{%
  \clearpage
  \if@twoside\ifodd\c@page
      \hbox{}\newpage\if@twocolumn\hbox{}\newpage\fi
  \fi\fi
}
```

```
\newcommand*{\cleardoubleevenpageusingstyle}[1]{\clearpage
  {\pagestyle{#1}\cleardoubleevenstandardpage}}
\newcommand*{\cleardoubleevenemptypage}{\cleardoubleevenpageusingstyle{empty}}
\newcommand*{\cleardoubleevenplainpage}{\cleardoubleevenpageusingstyle{plain}}
\newcommand*{\cleardoublestandardpage}{\cleardoubleoddstandardpage}
\renewcommand*{\cleardoublepage}{%
  \expandafter\ifnum \@nameuse{scr@v@2.98c}<\scr@compatibility\relax
    \cleardoubleemptypage
  \else
    \cleardoublestandardpage
  \fi
}
\newcommand*{\cleardoubleoddpage}{\cleardoubleoddemptypage}
\newcommand*{\cleardoubleevenpage}{\cleardoubleevenemptypage}
\newcommand*{\cleardoublepageusingstyle}[1]{\clearpage
  {\pagestyle{#1}\cleardoublestandardpage}}
\newcommand*{\cleardoubleemptypage}{\cleardoublepageusingstyle{empty}}
\newcommand*{\cleardoubleplainpage}{\cleardoublepageusingstyle{plain}}
⟨*class⟩
\KOMA@DeclareDeprecatedOption{cleardoubleempty}{cleardoublepage=empty}
\KOMA@DeclareDeprecatedOption{cleardoubleplain}{cleardoublepage=plain}
\KOMA@DeclareDeprecatedOption{cleardoublestandard}{cleardoublepage=current}
⟨/class⟩
```

9.3 Befehle für den Kompatibilitätsmodus

\headincludeon
\headincludeoff
\headseplineon
\headseplineoff
\footincludeon
\footincludeoff
\footseplineon
\footseplineoff

Da die KOMA-Script-Klassen ohnehin nicht mehr auf Kompatibilität zu Script 2.0 überprüft werden, wurde der Kompatibilitätsmodus ersatzlos gestrichen. Dazu gehören nach über zehn Jahren nun auch diese Anweisungen. Ehemals waren sie wie folgt definiert:

```
\if@compatibility
  \newcommand*\headincludeon{\@hincltrue}
  \newcommand*\headincludeoff{\@hinclfalse}
  \newcommand*\headseplineon{\@hsltrue \@hincltrue}
  \newcommand*\headseplineoff{\@hslfalse \@hinclfalse}
  \newcommand*\footincludeon{\@fincltrue}
  \newcommand*\footincludeoff{\@finclfalse}
  \newcommand*\footseplineon{\@fsltrue \@fincltrue}
  \newcommand*\footseplineoff{\@fslfalse \@finclfalse}
\fi
```

9.4 Definition der Standardseitenstile

\ps@plain
\ps@myheadings
\ps@headings

Es wird das Aussehen der Kopf- und Fußzeilen für die Standardseitenstile `empty`, `plain`, `headings` und `myheadings` festgelegt. Diese sind außerdem abhängig davon, ob es sich um einseitiges oder beidseitiges Layout handelt.

Im Gegensatz zu den Seitenstilen der Standardklassen steht die Seitennummer beim KOMA-Script Paket immer in der Fußzeile. Außerdem sind Trennlinien zwischen Kopf- und Textbereich sowie zwischen Text- und Fußbereich möglich.

Die Kopfzeile wird auch nicht mehr in Großbuchstaben gewandelt.

\set@tempdima@hw

Um nicht ständig das Gleiche schreiben zu müssen wird hier ein internes Makro verwendet.

```
\newcommand*{\set@tempdima@hw}{%
  \setlength{\@tempdima}{\textwidth}%
  \if@mincl
    \addtolength{\@tempdima}{\marginparsep}%
    \addtolength{\@tempdima}{\marginparwidth}%
  \fi
}
```

\pnumfont
\headfont
\footfont

Kopf- und Fußzeile sowie die Seitennummer werden in einer speziellen Schriftart gesetzt. Die drei Makros hier sind jedoch als interne Makros zu betrachten.

```
\newcommand*{\pnumfont}{\normalfont\normalcolor}
\newcommand*{\headfont}{\normalfont\normalcolor\slshape}
\newcommand*{\footfont}{}
```

pagenumber
pagination
pageheadfoot
pagehead
\scr@fnt@wrn@pagehead
pagefoot
\scr@fnt@wrn@pagefoot

Hier werden die Elemente definiert, deren Schriftart dann geändert werden kann.

```
\newcommand*{\scr@fnt@pagenumber}{\pnumfont}
\aliaskomafont{pagination}{pagenumber}
\newcommand*{\scr@fnt@pageheadfoot}{\headfont}
\aliaskomafont{pagehead}{pageheadfoot}
⟨*letter⟩
\newcommand*{\scr@fnt@wrn@pagehead}[1]{%
  ‘pagehead’ is only an alias of ‘pageheadfoot’.\MessageBreak
  Font of page foot will also be changed%
}
\newcommand*{\scr@fnt@pagefoot}{\footfont}
⟨/letter⟩
⟨*!letter⟩
\newcommand*{\scr@fnt@pagefoot}{\footfont}
\newcommand*{\scr@fnt@wrn@pagefoot}[1]{%
  This class does not use font element ‘pagefoot’!\MessageBreak
  Maybe you should load package ‘scrpage2’, before\MessageBreak
  using this element%
}
⟨/!letter⟩
```

\pagemark

Das ist die Seitenmarke, die nun für alle Klassen verwendet wird. Damit wird es leichter, fancyhdr statt scrpage2 zu verwenden.

```
\newcommand*{\pagemark}{{\usekomafont{pagenumber}{%
⟨letter⟩     \pagename\ %
    \thepage}}}
```

Zurück zur Definition der Seitenstile. Hier gibt es erhebliche Unterschiede zwischen der Briefklasse und den anderen Klassen. Das beginnt schon damit, dass es spezielle Stile für doppelseitige Briefe nicht gibt. Hinweis: Die *even*-Varianten werden im einseitigen Satz vom LaTeX-Kern selbst nicht verwendet. Es schadet also nicht, sie trotzdem identisch zum doppelseitigen Satz zu definieren.

```
⟨*!letter⟩
\renewcommand*{\ps@plain}{%
  \renewcommand*{\@oddhead}{}%
  \let\@evenhead\@oddhead
  \renewcommand*{\@evenfoot}{%
    \set@tempdima@hw\hss\hb@xt@ \@tempdima{\vbox{%
        \if@fsl \hrule \vskip 3\p@ \fi
        \hb@xt@ \@tempdima{{\pagemark\hfil}}}}}%
  \renewcommand*{\@oddfoot}{%
    \set@tempdima@hw\hb@xt@ \@tempdima{\vbox{%
        \if@fsl \hrule \vskip 3\p@ \fi
        \hb@xt@ \@tempdima{{\hfil\pagemark
            \if@twoside\else\hfil\fi}}}}\hss}%
}%
\newcommand*{\ps@headings}{\let\@mkboth\markboth
  \renewcommand*{\@evenhead}{%
    \set@tempdima@hw\hss\hb@xt@ \@tempdima{\vbox{%
        \hb@xt@ \@tempdima{{\headfont\strut\leftmark\hfil}}%
        \if@hsl \vskip 1.5\p@ \hrule \fi}}}%
  \renewcommand*{\@oddhead}{%
    \set@tempdima@hw\hb@xt@ \@tempdima{\vbox{%
        \hb@xt@ \@tempdima{{\headfont\hfil\strut\rightmark
            \if@twoside\else\hfil\fi}}%
        \if@hsl \vskip 1.5\p@ \hrule \fi}}\hss}%
  \renewcommand*{\@evenfoot}{%
    \set@tempdima@hw\hss\hb@xt@ \@tempdima{\vbox{%
        \if@fsl \hrule \vskip 3\p@ \fi
        \hb@xt@ \@tempdima{{\pagemark\hfil}}}}}%
  \renewcommand*{\@oddfoot}{%
    \set@tempdima@hw\hb@xt@ \@tempdima{\vbox{%
        \if@fsl \hrule \vskip 3\p@ \fi
        \hb@xt@ \@tempdima{{\hfil\pagemark
            \if@twoside\else\hfil\fi}}}}\hss}%
⟨*article⟩
  \renewcommand*{\sectionmark}[1]{%
    \if@twoside\expandafter\markboth\else\expandafter\markright\fi
    {\ifnum \c@secnumdepth >\z@%
      \sectionmarkformat\fi ##1}{}}%
  \renewcommand*{\subsectionmark}[1]{%
```

```
    \if@twoside
      \markright{\ifnum \c@secnumdepth >\@ne%
        \subsectionmarkformat\fi ##1}%
    \fi
  }%
⟨/article⟩
⟨*report | book⟩
  \renewcommand*{\chaptermark}[1]{%
    \if@twoside\expandafter\markboth\else\expandafter\markright\fi
    {\ifnum \c@secnumdepth >\m@ne
⟨book⟩         \if@mainmatter
           \chaptermarkformat
⟨book⟩         \fi
      \fi
      ##1}{}%
  }%
  \renewcommand*{\sectionmark}[1]{%
    \if@twoside
      \markright{\ifnum \c@secnumdepth >\z@
        \sectionmarkformat\fi
        ##1}%
    \fi
  }%
⟨/report | book⟩
}%
\newcommand*{\ps@myheadings}{\let\@mkboth\@gobbletwo
  \renewcommand*{\@evenhead}{%
    \set@tempdima@hw\hss\hb@xt@ \@tempdima{\vbox{%
        \hb@xt@ \@tempdima{{\headfont\strut\leftmark\hfil}}%
        \if@hsl \vskip 1.5\p@ \hrule \fi}}}%
  \renewcommand*{\@oddhead}{%
    \set@tempdima@hw\hb@xt@ \@tempdima{\vbox{%
        \hb@xt@ \@tempdima{{\headfont\hfil\strut\rightmark
            \if@twoside\else\hfil\fi}}%
        \if@hsl \vskip 1.5\p@ \hrule \fi}}\hss}%
  \renewcommand*{\@evenfoot}{%
    \set@tempdima@hw\hss\hb@xt@ \@tempdima{\vbox{%
        \if@fsl \hrule \vskip 3\p@ \fi
        \hb@xt@ \@tempdima{{\pagemark\hfil}}}}}%
  \renewcommand*{\@oddfoot}{%
    \set@tempdima@hw\hb@xt@ \@tempdima{\vbox{%
        \if@fsl \hrule \vskip 3\p@ \fi
        \hb@xt@ \@tempdima{{\hfil\pagemark
            \if@twoside\else\hfil\fi}}}\hss}%
⟨!article⟩   \renewcommand*{\chaptermark}[1]{}%
⟨article⟩   \renewcommand*{\subsectionmark}[1]{}%
  \renewcommand*{\sectionmark}[1]{}%
}
⟨/!letter⟩
```

```
⟨*letter⟩
\renewcommand*{\ps@plain}{%
  \renewcommand*{\@oddhead}{%
    \vbox{\vbox{\hsize=\textwidth\hbox to\textwidth{%
          \parbox[b]{\textwidth}{\strut
            \ifnum\@pageat>-1
              \ifnum\@pageat<3
                \ifcase\@pageat\raggedright\or\centering\or\raggedleft\fi
                \pagemark
              \else
                \hfill
              \fi
            \else
              \hfill
            \fi
          }%
        }%
        \if@hsl\kern1pt\rule{\textwidth}{.4pt}\fi
      }%
    }%
  }%
  \let\@evenhead\@oddhead%
  \renewcommand*{\@oddfoot}{%
    \parbox[t]{\textwidth}{%
      \if@fsl
        {%
          \raggedright%
          \vskip-\baselineskip\vskip.4pt
          \hrulefill\\
        }%
      \fi
      \ifnum\@pageat>2
        \ifcase\@pageat\or\or\or\raggedright\or\centering\or\raggedleft\fi
        \strut\pagemark
      \else
        \hfill
      \fi
    }%
  }%
  \let\@evenfoot\@oddfoot
}
\newcommand*{\ps@headings}{\let\@mkboth\markboth
  \renewcommand*{\@oddhead}{%
    \vbox{%
      \vbox{\hsize=\textwidth\hbox to\textwidth{\headfont\noindent
          \usekomavar{nexthead}}}%
      \if@hsl\kern1pt\rule{\textwidth}{.4pt}\fi%
    }%
  }%
```

```
  \let\@evenhead\@oddhead
  \renewcommand*{\@oddfoot}{%
    \parbox[t]{\textwidth}{%
      \if@fsl
        {%
          \raggedright%
          \vskip-\baselineskip\vskip.4pt
          \hrulefill\\
        }%
      \fi
      \vbox{\hsize=\textwidth\hbox to\textwidth{\headfont\footfont\noindent
          \usekomavar{nextfoot}}}%
    }%
  }%
  \let\@evenfoot\@oddfoot
}
\newcommand*{\ps@myheadings}{%
  \ps@headings
  \let\@mkboth\@gobbletwo
}
⟨/letter⟩
```

9.5 Festlegung des Seitenstils auf besonderen Seiten

So etwas gibt es derzeit nicht für Briefe.

`\titlepagestyle` `\partpagestyle` `\chapterpagestyle` `\indexpagestyle`

Auf verschiedenen Seiten wird automatisch mit `\thispagestyle` auf einen anderen Seitenstil umgeschaltet. Welcher das ist, ist in diesem Makros abgelegt und kann bei Bedarf geändert werden. Voreingestellt ist der in früheren Versionen fest verdrahtete Seitenstil `plain`.

```
\newcommand*{\titlepagestyle}{plain}
⟨book | report⟩\newcommand*{\partpagestyle}{plain}
⟨book | report⟩\newcommand*{\chapterpagestyle}{plain}
⟨class⟩\newcommand*{\indexpagestyle}{plain}
```

9.6 Standardeinstellungen

Der voreingestellte Seitenstil hängt von der verwendeten Klasse ab:

```
⟨report | article | letter⟩\pagestyle{plain}
⟨book⟩\pagestyle{headings}
```

Die Nummerierung erfolgt hingegen immer mit arabischen Zahlen:

```
\pagenumbering{arabic}
```

Im zweiseitigen Satz wird die letzte Zeile bündig gesetzt, im einseitigen und bei Briefen generell jedoch nicht:

```
⟨*!letter⟩
\if@twoside
  \flushbottom
```

```
\else
⟨/!letter⟩
  \raggedbottom
⟨!letter⟩\fi
```

Im zweispaltigen Satz wird `\sloppy` verwendet und die letzte Zeile jeweils bündig gesetzt. Briefe sind einspaltig.

```
⟨*!letter⟩
\if@twocolumn
  \twocolumn
  \sloppy
  \flushbottom
\else
⟨/!letter⟩
  \onecolumn
⟨!letter⟩\fi
```

Datei j
scrkpar.dtx

10 Absatzformatierung und -umbruch

In diesen Bereich gehört alles, was die Absatzformatierung selbst betrifft. Dazu kommen dann noch einige Kleinigkeiten aus dem Seitenumbruch.

10.1 Optionen für das Absatzlayout

Ab Version 2.8i können wir auch mit Absatzlayouts umgehen, die einen Absatzabstand statt einem Absatzeinzug verlangen. Gesteuert wird dies über Optionen und Schalter.

`\setparsizes` Über dieses Makro wird die Änderung der Absatzparameter `\parskip`, `\parindent`, `\parfillskip` gesetzt. Das erste Argument ist der Einzug, das zweite der Abstand und das dritte die Füllung. Aktiviert werden die Änderungen wie bei `\fontsize` über `\selectfont`. Obwohl in der Voreinstellung absolut gearbeitet wird, wird hier intern `\par@updaterelative` verwendet.

```
\newcommand*{\setparsizes}[3]{%
  \edef\f@parindent{\the\parindent}%
  \edef\f@parskip{\the\parskip}%
  \edef\f@parfillskip{\the\parfillskip}%
  \def\scr@parindent{#1}%
  \def\scr@parskip{#2}%
  \def\scr@parfillskip{#3}%
  \def\par@update{\let\par@update\default@par@update\par@updaterelative}%
}
```

`\f@parindent` Eingestellter Absatzeinzug.

`\f@parskip` Eingestellter Absatzabstand.

`\f@parfillskip` Eingestellte Absatzfüllung.

Diese drei Werte werden automatisch bei der Font-Initialisierung eingestellt und sind vorher ungültig!

```
\newcommand*{\f@parindent}{\the\parindent}
\newcommand*{\f@parskip}{\the\parskip}
\newcommand*{\f@parfillskip}{\the\parfillskip}
\AtEndOfClass{%
  \edef\f@parindent{\the\parindent}%
  \edef\f@parskip{\the\parskip}%
  \edef\f@parfillskip{\the\parfillskip}%
}
```

\par@update Dieses Makro wird später in \selectfont die Änderung vornehmen.

\default@par@update In der Voreinstellung findet keine Änderung statt. Dies wird jedoch durch die Auswahl einer entsprechenden Option geändert.

```
\newcommand*{\par@update}{}
\let\par@update\relax
\newcommand*{\default@par@update}{}
\let\default@par@update\relax
```

parskip, parskip-, parskip+, parskip*, halfparskip, halfparskip-, halfparskip+, halfparskip* Diese neun Optionen steuern die Umschaltung zwischen den Modi. Dabei schalten alle parskip-Optionen einen Absatzabstand ein, wohingegen die parindent-Option den Absatzeinzug einschaltet. Die +-Variante sorgt außerdem dafür, dass die letzte Zeile eines Absatzes maximal zu zwei Dritteln gefüllt wird. Entsprechend sorgt die *-Variante für eine maximale Füllung von drei Vierteln. Die normale Variante sorgt lediglich für einen freien Raum von 1 em. Die -Variante sorgt für überhaupt nichts.

\scr@parindent, parindent Der einzustellende Absatzeinzug.

\scr@parskip Der einzustellende Absatzabstand.

\scr@parfillskip Die einzustellende Absatzfüllung.

```
\newcommand*{\scr@parindent}{1em}
\newcommand*{\scr@parskip}{\z@}
\newcommand*{\scr@parfillskip}{\z@ \@plus 1fil}
```

Neu bei KOMA-Script-3 ist, dass die Optionen wie bei scrlttr2 über eine eingzige keyval gesetzt werden können.

```
\KOMA@key{parskip}[true]{%
  \begingroup
    \KOMA@set@ncmdkey{parskip}{@tempa}{%
      {never}{0},%
      {false}{1},{off}{1},{no}{1},%
      {full-}{2},%
      {half-}{3},%
      {full}{4},{true}{4},{on}{4},{yes}{4},%
      {half}{5},%
      {full+}{6},%
      {half+}{7},%
      {full*}{8},%
      {half*}{9},%
      {relative}{10},%
      {absolute}{11}%
    }{#1}%
    \ifcase\number\@tempa% 0
      \endgroup
      \setparsizes{1em}{\z@}{\z@ \@plus 1fil}%
      \if@atdocument\AfterKOMAoptions{\selectfont}\fi
    \or% 1
```

```
    \endgroup
    \setparsizes{1em}{\z@ \@plus \p@}{\z@ \@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 2
    \endgroup
    \setparsizes{\z@}{\baselineskip \@plus .1\baselineskip}{%
      \z@ \@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 3
    \endgroup
    \setparsizes{\z@}{.5\baselineskip \@plus .5\baselineskip}{%
      \z@ \@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 4
    \endgroup
    \setparsizes{\z@}{\baselineskip \@plus .1\baselineskip}{%
      1em \@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 5
    \endgroup
    \setparsizes{\z@}{.5\baselineskip \@plus .5\baselineskip}{%
      1em \@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 6
    \endgroup
    \setparsizes{\z@}{\baselineskip \@plus .1\baselineskip}{%
      .3333\linewidth\@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 7
    \endgroup
    \setparsizes{\z@}{.5\baselineskip \@plus .5\baselineskip}{%
      .3333\linewidth \@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 8
    \endgroup
    \setparsizes{\z@}{\baselineskip \@plus .1\baselineskip}{%
      .25\linewidth \@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 9
    \endgroup
    \setparsizes{\z@}{.5\baselineskip \@plus .5\baselineskip}{%
      .25\linewidth \@plus 1fil}%
    \if@atdocument\AfterKOMAoptions{\selectfont}\fi
  \or% 10
    \endgroup
    \ifx\par@updaterelative\undefined
      \expandafter\AtEndOfClass
    \else
      \expandafter\@firstofone
```

```
      \fi
      {%
        \ifx\par@update\default@par@update
          \let\par@update\par@updaterelative
        \fi
        \let\default@par@update=\par@updaterelative
      }%
    \or%11
      \endgroup
      \ifx\par@updaterelative\undefined
        \expandafter\AtEndOfClass
      \else
        \expandafter\@firstofone
      \fi
      {%
        \ifx\par@update\default@par@update
          \let\par@update\relax
        \fi
        \let\default@par@update=\relax
      }%
    \else% sonst
      \endgroup
  \fi
}
\KOMA@DeclareDeprecatedOption{parskip-}{parskip=full-}
\KOMA@DeclareDeprecatedOption{parskip+}{parskip=full+}
\KOMA@DeclareDeprecatedOption{parskip*}{parskip=full*}
\KOMA@DeclareDeprecatedOption{halfparskip}{parskip=half}
\KOMA@DeclareDeprecatedOption{halfparskip-}{parskip=half-}
\KOMA@DeclareDeprecatedOption{halfparskip+}{parskip=half+}
\KOMA@DeclareDeprecatedOption{halfparskip*}{parskip=half*}
\KOMA@DeclareDeprecatedOption{parindent}{parskip=false}
```

Geplant!

Hierfür sollte eine keyval-Option mit Werten definiert werden. Dazu dann neue Werte für absolutes oder relatives Verhalten.

10.2 Abatzformatierung

`lineskip`
`normallineskip`

Minimaler Zeilenabstand:

```
\setlength{\lineskip}{\p@}
\setlength{\normallineskip}{\p@}
```

`columnsep`
`columnseprule`

Spaltenabstand und Spaltentrennlinie (nicht bei Briefen):

```
⟨*!letter⟩
\setlength{\columnsep}{10\p@}
\setlength{\columnseprule}{\z@}
⟨/!letter⟩
```

\selectfont

Spätestens ab Version 3.0 soll die Möglichkeit bestehen, \parskip, \parindent und \parfillskip mit der Schriftgröße automatisch anzupassen. Dazu muss \selectfont entsprechend erweitert werden.

\scr@selectfont
\scr@new@selectfont

Damit das auch bei Verwendung des everysel-Pakets funktioniert, wird hier zusätzliche Vorsorge getroffen.

```
\newcommand*{\scr@selectfont}{}\let\scr@selectfont\selectfont
\BeforePackage{everysel}{%
  \ifx\selectfont\scr@new@selectfont\else
    \ClassWarningNoLine{\KOMAClassName}{discard change of \string\selectfont}%
  \fi
  \let\selectfont\scr@selectfont
}
\AfterPackage{everysel}{%
  \g@addto@macro\selectfont{\par@update}%
  \let\scr@new@selectfont\selectfont
}
\g@addto@macro\selectfont{\par@update}
\newcommand*{\scr@new@selectfont}{}\let\scr@new@selectfont\selectfont
```

\par@updaterelative

Die eigentliche Änderung verbirgt sich in \par@updaterelative. Ggf. wird \par@update zu \par@updaterelative.

```
\newcommand*{\par@updaterelative}{%
```

Die neuen Werte werden nur gesetzt, wenn die bisherigen Werten den erwarteten Werten entsprechen. Sonst lassen wir lieber die Finger davon, weil wir dann davon ausgehen, dass der Anwender die so setzen wollte.

```
  \begingroup
    \edef\@tempa{\the\parindent}\ifx\@tempa\f@parindent
      \aftergroup\parindent@update
⟨*trace⟩
    \else
      \ClassInfo{\KOMAClassName}{\string\parindent\space not changed}%
⟨/trace⟩
    \fi
    \edef\@tempa{\the\parskip}\ifx\@tempa\f@parskip
      \aftergroup\parskip@update
⟨*trace⟩
    \else
      \ClassInfo{\KOMAClassName}{\string\parskip\space not changed}%
⟨/trace⟩
    \fi
    \edef\@tempa{\the\parfillskip}\ifx\@tempa\f@parfillskip
      \aftergroup\parfillskip@update
⟨*trace⟩
    \else
      \ClassInfo{\KOMAClassName}{\string\parfillskip\space not changed}%
⟨/trace⟩
    \fi
```

```
  \endgroup
}
```

\parindent@update
\parskip@update
\parfillskip@update

Ein paar Hilfsmakros.

```
\newcommand*{\parindent@update}{%
  \scr@defaultunits\parindent\scr@parindent
  \begingroup
    \let\@tempb\endgroup
    \edef\@tempa{\the\parindent}\ifx\@tempa\f@parindent\else
      \def\@tempb{\endgroup\edef\f@parindent{\the\parindent}}%
⟨trace⟩       \ClassInfo{\KOMAClassName}{\string\parindent=\the\parindent}%
    \fi
  \@tempb
}
\newcommand*{\parskip@update}{%
  \scr@defaultunits\parskip\scr@parskip
  \begingroup
    \let\@tempb\endgroup
    \edef\@tempa{\the\parskip}\ifx\@tempa\f@parskip\else
      \def\@tempb{\endgroup\edef\f@parskip{\the\parskip}}%
⟨trace⟩       \ClassInfo{\KOMAClassName}{\string\parskip=\the\parskip}%
    \fi
  \@tempb
}
\newcommand*{\parfillskip@update}{%
  \scr@defaultunits\parfillskip\scr@parfillskip
  \begingroup
    \let\@tempb\endgroup
    \edef\@tempa{\the\parfillskip}\ifx\@tempa\f@parfillskip\else
      \def\@tempb{\endgroup\edef\f@parfillskip{\the\parfillskip}}%
⟨trace⟩       \ClassInfo{\KOMAClassName}{\string\parfillskip=\the\parfillskip}%
    \fi
  \@tempb
}
```

\scr@defaultunits
\scr@@defaultunits
\scr@@@defaultunits

Damit `\par@updaterelative` überhaupt funktionieren kann, wird `\scr@defaultunits` benötigt. Dieses Makro arbeitet prinzipiell wie `\@defaultunits` bekommt aber Dimension bzw. Skip als erstes und den Wert als zweites Argument. Als Besonderheit dürften im Wert auch andere Dimensions bzw. Skips vor und nach `plus` und `minus` verwendet werden. Es sind also auch Angaben der Art „`12 plus 1 minus 2`“ sowie „`\baselineskip\@plus.1\baselineskip`“ gültig.

```
\newcommand*{\scr@defaultunits}[2]{%
  \begingroup
    \edef\@tempa{#2}%
    \expandafter\scr@@defaultunits\expandafter#1\@tempa plusplus\@nnil
    \edef\@tempa{\noexpand\endgroup\noexpand\setlength{\noexpand#1}{\the#1}}%
  \@tempa
```

```
}
\newcommand*{\scr@@defaultunits}{}
\def\scr@@defaultunits#1#2plus#3plus#4\@nnil{%
  \ifx\relax#3\relax
    \scr@@@defaultunits#1{}#2minusminus\@nnil
  \else
    \scr@@@defaultunits#1{#2}#3minusminus\@nnil
  \fi
}
\newcommand*{\scr@@@defaultunits}{}
\def\scr@@@defaultunits#1#2#3minus#4minus#5\@nnil{%
  \ifx\relax#2\relax
    \@defaultunits\@tempskipa#3pt\relax\@nnil
    \setlength{#1}{\@tempskipa}%
  \else
    \@defaultunits\@tempskipa\z@\@plus#3pt\relax\@nnil
    \setlength{#1}{\@tempskipa}%
    \@defaultunits\@tempskipa#2pt\relax\@nnil
    \addtolength{#1}{\@tempskipa}%
  \fi
  \ifx\relax#4\relax\else
    \@defaultunits\@tempskipa\z@\@minus #4pt\relax\@nnil
    \addtolength{#1}{\@tempskipa}%
  \fi
}
```

Absatzabstand und Absatzeinzug:

`\@list@extra`
`\add@extra@listi`

Ab Version 2.8i wird hier optionsabhängig gearbeitet. Dabei müssen auch die Befehle bei der Umschaltung der Schriftgröße für `\small` und `\footnotesize` geändert werden.

```
\newcommand*{\@list@extra}{%
  \ifdim\parskip>\z@\topsep\z@\parsep\parskip\itemsep\z@\fi
}
\newcommand*{\add@extra@listi}[1]{%
  \expandafter\let\csname #1@listi\endcsname=\@listi
  \def\@listi{\csname #1@listi\endcsname\@list@extra}%
}
```

`\@listi`
`\@listI`
`\@listii`
`\@listiii`
`\footnotesize`
`\small`

```
⟨10pt | 11pt | 12pt⟩\@ifundefined{@list@extra}{}{%
  \l@addto@macro{\@listi}{\@list@extra}%
  \let\@listI=\@listi
  \l@addto@macro{\@listii}{\@list@extra}%
  \l@addto@macro{\@listiii}{\@list@extra}%
  \l@addto@macro{\footnotesize}{\protect\add@extra@listi{ftns}}%
  \l@addto@macro{\small}{\protect\add@extra@listi{sml}}%
⟨10pt | 11pt | 12pt⟩}
⟨10pt | 11pt | 12pt⟩\@listi
```

10.3 Umbruchsteuerung

Für die Umbruchsteuerung sind einige Penalties zuständig. Diese sind im LaTeX-Kern definiert. Leider sind `\@lowpenalty`, `\@medpenalty` und `\@highpenalty` aber nicht mit Voreinstellungen versehen, besitzen einheitlich die Voreinstellung 0. Hier werden deshalb die Werte aus den Standardklassen übernommen:

```
\@lowpenalty  = 51
\@medpenalty  =151
\@highpenalty =301
```

Datei k
scrktitl.dtx

11 Die Titelei

Die Titelei ist gegenüber den Standardklassen erheblich erweitert. Trotzdem kann ein Titel einer Standardklasse unmittelbar mit einer KOMA-Script-Klasse gesetzt werden. Briefe besitzen allerdings keine Titelei im eigentlichen Sinne.

11.1 Optionen der Titelei

`titlepage`
`notitlepage`

Es werden zei Arten von Titeln unterstützt. Da sind zum einen eigene Titelseiten, zum anderen sogenannte Titelköpfe, also Titel, die nicht eine eigene Seite erhalten, sondern am Anfang einer Seite stehen. Zwischen den beiden Arten wird mit einer Option umgeschaltet.

`\if@titlepage`
`\@titlepagetrue`
`\@titlepagefalse`

Die gewählte Einstellung wird in einem Schalter gespeichert.

```
⟨*class⟩
\newif\if@titlepage
⟨article⟩\@titlepagefalse
⟨report | book⟩\@titlepagetrue
⟨/class⟩
```

Im Falle von scrextend ist die Option Teil der Erweiterung.

```
⟨*package & extend⟩
\scr@ext@activateable{title}{%
  \scr@ifundefinedorrelax{if@titlepage}{%
    \expandafter\newif\csname if@titlepage\endcsname
    \@titlepagefalse
  }{}
⟨/package & extend⟩
```

```
⟨*class | package⟩
\KOMA@ifkey{titlepage}{@titlepage}%
⟨*extend⟩
```

Diese Erweiterung soll dann auch tatsächlich sofort aktiviert werden:

```
  \def\scr@ext@immediate@title{%
    \scr@ext@activate{title}%
    \let\scr@ext@immediate@title\relax
  }%
}
⟨/extend⟩
⟨/class | package⟩
\KOMA@DeclareStandardOption%
⟨package⟩   [scrextend]%
```

```
{notitlepage}{titlepage=false}
```

abstract
abstracton
abstractoff

Obwohl die Zusammenfassung nicht unbedingt auf einer Titelseite steht, gehört sie doch zur Titelei und wird daher hier mit behandelt. Die Zusammenfassung kann mit einem standardmäßigen Titel versehen werden. Der Titel kann aber auch unterdrückt werden.

Bei Büchern gibt es keine Zusammenfassung als gesonderte Umgebung. Als Ersatz kann bei Büchern auf `\addchap*` zurückgegriffen werden.

\if@abstrt
\@abstrttrue
\@abstrtfalse

Die Enscheidung, ob der Titel der Zusammenfassung gesetzt werden soll, wird in einem Schalter gespeichert.

```
\newif\if@abstrt\@abstrtfalse
```

```
\KOMA@ifkey{abstract}{@abstrt}
\KOMA@DeclareStandardOption{abstracton}{abstract=true}
\KOMA@DeclareStandardOption{abstractoff}{abstract=false}
```

11.2 Definitionen der Titelei

\extratitle
\@extratitle
\titlehead
\@titlehead
\subject
\@subject
\subtitle
\@subtitle
\publishers
\@publishers
\uppertitleback
\@uppertitleback
\lowertitleback
\@lowertitleback
\dedication
\@dedication

Da der Titel im KOMA-Script Paket wesentlich mehr Angaben erlaubt als bei den Standardklassen, gibt es natürlich auch einige Befehle, mit denen diese gesetzt werden können. Alle zusätzlichen Angaben sind optional und können auch weggelassen werden. In diesem Fall werden Leerfelder verwendet. Bis auf `\subject` sind alle Zusatzbefehle `\long` deklariert, können also ganze Absätze enthalten.

```
⟨package & extend⟩\scr@ext@addto@activateable{title}{%
\newcommand*{\@extratitle}{}%
\newcommand{\extratitle}[1]{\gdef\@extratitle{%
⟨extend⟩     ##1%
⟨!extend⟩     #1%
}}%
\newcommand*{\@titlehead}{}%
\newcommand{\titlehead}[1]{\gdef\@titlehead{%
⟨extend⟩     ##1%
⟨!extend⟩     #1%
}}%
\newcommand*{\@subject}{}%
\newcommand*{\subject}[1]{\gdef\@subject{%
⟨extend⟩     ##1%
⟨!extend⟩     #1%
}}%
\newcommand*{\subtitle}[1]{\gdef\@subtitle{%
⟨extend⟩     ##1%
⟨!extend⟩     #1%
}}%
\newcommand*{\@subtitle}{}%
\newcommand*{\@publishers}{}%
\newcommand{\publishers}[1]{\gdef\@publishers{%
```

```
⟨extend⟩     ##1%
⟨!extend⟩     #1%
}}%
\newcommand*{\@uppertitleback}{}%
\newcommand{\uppertitleback}[1]{\gdef\@uppertitleback{%
⟨extend⟩     ##1%
⟨!extend⟩     #1%
}}%
\newcommand*{\@lowertitleback}{}%
\newcommand{\lowertitleback}[1]{\gdef\@lowertitleback{%
⟨extend⟩     ##1%
⟨!extend⟩     #1%
}}%
\newcommand*{\@dedication}{}%
\newcommand{\dedication}[1]{\gdef\@dedication{%
⟨extend⟩     ##1%
⟨!extend⟩     #1%
}}%
```

`\next@tpage` Innerhalb eines Titels auf die nächste Seite umschalten.

```
\newcommand*{\next@tpage}{\clearpage\thispagestyle{empty}}%
```

`\maketitle` Mit dieser Anweisung wird der Titel generiert, dessen Inhalt zuvor gesetzt wurde. Es muss unterschieden werden, ob der Titel auf einer eigenen Titelseite oder als Seitenkopf erstellt werden soll. Der KOMA-Script-Titel ist u. U. sehr groß. Im Fall dass diverse Zusatzelemente verwendet werden, sollte eigentlich immer eine Titelseite verwendet werden.

```
⟨package & extend⟩\let\maketitle\relax\let\@maketitle\relax
\newcommand*\maketitle[1][1]{%
  \if@titlepage
    \begin{titlepage}
      \setcounter{page}{%
⟨extend⟩         ##1%
⟨!extend⟩         #1%
      }%
      \let\footnotesize\small
      \let\footnoterule\relax
      \let\footnote\thanks
      \renewcommand*\thefootnote{\@fnsymbol\c@footnote}%
      \let\@oldmakefnmark\@makefnmark
      \renewcommand*{\@makefnmark}{\rlap\@oldmakefnmark}%
      \ifx\@extratitle\@empty \else
        \noindent\@extratitle\next@tpage\cleardoubleemptypage
        \thispagestyle{empty}%
      \fi
⟨!extend⟩       \setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
⟨extend⟩       \parskip\z@ \parindent\z@ \parfillskip\z@\@plus 1fil
      \ifx\@titlehead\@empty \else
        \begin{minipage}[t]{\textwidth}%
```

```
        \@titlehead
        \end{minipage}\par
      \fi
      \null\vfill
      \begin{center}
        \ifx\@subject\@empty \else
          {\subject@font \@subject \par}%
          \vskip 3em
        \fi
        {\titlefont\huge \@title\par}%
        \vskip 1em
        {\ifx\@subtitle\@empty\else\usekomafont{subtitle}\@subtitle\par\fi}%
        \vskip 2em
        {\Large \lineskip 0.75em
          \begin{tabular}[t]{c}
            \@author
          \end{tabular}\par
        }%
        \vskip 1.5em
        {\Large \@date \par}%
        \vskip \z@ \@plus3fill
        {\Large \@publishers \par}%
        \vskip 3em
      \end{center}\par
      \@thanks
      \vfill\null
      \if@twoside\next@tpage
        \begin{minipage}[t]{\textwidth}
          \@uppertitleback
        \end{minipage}\par
        \vfill
        \begin{minipage}[b]{\textwidth}
          \@lowertitleback
        \end{minipage}
      \fi
      \ifx\@dedication\@empty \else
        \next@tpage\null\vfill
        {\centering \Large \@dedication \par}%
        \vskip \z@ \@plus3fill
        \if@twoside \next@tpage\cleardoubleemptypage \fi
      \fi
    \end{titlepage}

  \else
    \par
    \@tempcnta=%
⟨extend⟩     ##1%
⟨!extend⟩     #1%
    \relax\ifnum\@tempcnta=1\else
⟨class⟩       \ClassWarning{\KOMAClassName}{%
```

```
⟨package⟩      \PackageWarning{scrextend}{%
        Optional argument of \string\maketitle\space ignored
        at\MessageBreak
        notitlepage-mode%
      }%
    \fi
    \begingroup
      \renewcommand*\thefootnote{\@fnsymbol\c@footnote}%
      \let\@oldmakefnmark\@makefnmark
      \renewcommand*{\@makefnmark}{\rlap\@oldmakefnmark}
      \if@twocolumn
        \ifnum \col@number=\@ne
          \@maketitle
        \else
          \twocolumn[\@maketitle]%
        \fi
      \else
        \newpage
        \global\@topnum\z@
        \@maketitle
      \fi
      \thispagestyle{\titlepagestyle}\@thanks
    \endgroup
  \fi
  \setcounter{footnote}{0}%
  \let\thanks\relax
  \let\maketitle\relax
  \let\@maketitle\relax
  \global\let\@thanks\@empty
  \global\let\@author\@empty
  \global\let\@date\@empty
  \global\let\@title\@empty
  \global\let\@subtitle\@empty
  \global\let\@extratitle\@empty
  \global\let\@titlehead\@empty
  \global\let\@subject\@empty
  \global\let\@publishers\@empty
  \global\let\@uppertitleback\@empty
  \global\let\@lowertitleback\@empty
  \global\let\@dedication\@empty
  \global\let\author\relax
  \global\let\title\relax
  \global\let\extratitle\relax
  \global\let\titlehead\relax
  \global\let\subject\relax
  \global\let\publishers\relax
  \global\let\uppertitleback\relax
  \global\let\lowertitleback\relax
  \global\let\dedication\relax
```

```
  \global\let\date\relax
  \global\let\and\relax
}%
```

\@maketitle Damit wird im Fall des Titelkopfes die eigentliche Arbeit geleistet.

```
\newcommand*{\@maketitle}{%
  \clearpage
  \let\footnote\thanks
  \ifx\@extratitle\@empty \else
    \noindent\@extratitle \next@tpage \if@twoside \null\next@tpage \fi
  \fi
⟨!extend⟩  \setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
⟨extend⟩  \parskip\z@ \parindent\z@ \parfillskip\z@\@plus 1fil
  \ifx\@titlehead\@empty \else
    \begin{minipage}[t]{\textwidth}
      \@titlehead
    \end{minipage}\par
  \fi
  \null
  \vskip 2em%
  \begin{center}%
    \ifx\@subject\@empty \else
      {\subject@font \@subject \par}
      \vskip 1.5em
    \fi
    {\titlefont\huge \@title \par}%
    \vskip .5em
    {\ifx\@subtitle\@empty\else\usekomafont{subtitle}\@subtitle\par\fi}%
    \vskip 1em
    {\Large
      \lineskip .5em%
      \begin{tabular}[t]{c}
        \@author
      \end{tabular}\par
    }%
    \vskip 1em%
    {\Large \@date \par}%
    \vskip \z@ \@plus 1em
    {\Large \@publishers \par}
    \ifx\@dedication\@empty \else
      \vskip 2em
      {\Large \@dedication \par}
    \fi
  \end{center}%
  \par
  \vskip 2em
}%
```

11.3 Umgebung für die Titelseite

`titlepage` Die Titelseite bedarf ebenfalls einer gesonderter Umgebung. Beim zweispaltigen Layout soll die Titelseite z. B. einspaltig gedruckt werden.

```
⟨package & extend⟩\scr@ifundefinedorrelax{titlepage}{%
\newenvironment{titlepage}{%
⟨report | book⟩   \cleardoublepage
  \if@twocolumn
    \@restonecoltrue\onecolumn
  \else
    \@restonecolfalse\newpage
  \fi
  \thispagestyle{empty}%
  \if@compatibility
    \setcounter{page}{0}%
  \fi
}{%
  \if@restonecol\twocolumn \else \newpage \fi
}%
⟨package & extend⟩}{}%
```

Im Fall von scrextend muss jetzt das Kürzel *title* beendet werden:

```
⟨package & extend⟩}\csname scr@ext@immediate@title\endcsname
```

11.4 Fonts für den Titel

`\titlefont` Wie oben zu sehen ist, wird in der Titelei ein eigenes Schriftmakro verwendet. Dieses ist als internes Makro zu verstehen. Der Anwender sollte stattdessen auf das entsprechende Element zugreifen (siehe unten).

`\subject@font` Ebenso verhält es sich mit den *Subject* im Titel. Hier ist aber von vornherein ein internes Makro definiert.

```
\newcommand*\titlefont{\sectfont}%
\newcommand*{\subject@font}{\normalfont\normalcolor\bfseries\Large}%
```

`subtitle`

```
\newkomafont{subtitle}{\usekomafont{title}\large}%
```

`\scr@fnt@title` `\scr@fnt@subject` Die beiden Elemente, deren Schrift geändert werden kann.

```
\newcommand*{\scr@fnt@title}{\titlefont}%
\newcommand*{\scr@fnt@subject}{\subject@font}%
```

11.5 Umgebung für die Zusammenfassung

`abstract` scrartcl und scrreprt bieten die Möglichkeit einer Zusammenfassung, eines sogenannten *Abstracts*. Wenn eine Titelseite verlangt wurde, wird auch

die Zusammenfassung auf eine eigene Seite gesetzt. scrbook kennt keine Zusammenfassung dieser Art. In Büchern werden Zusammenfassungen üblicherweise mit Kapitelcharakter gesetzt.

```
\newenvironment{abstract}{%
  \if@titlepage
    \titlepage
    \null\vfil
    \@beginparpenalty\@lowpenalty
    \if@abstrt
      \begin{center}
        \normalfont\sectfont\nobreak\abstractname
        \@endparpenalty\@M
      \end{center}
    \fi
  \else
    \if@twocolumn\if@abstrt
        \addsec*{\abstractname}
      \fi
    \else
      \if@abstrt
        \small
        \begin{center}
          {\normalfont\sectfont\nobreak\abstractname
            \vspace{-.5em}\vspace{\z@}}%
        \end{center}
      \fi
      \quotation
    \fi
  \fi
}{%
  \if@titlepage
    \par\vfil\null\endtitlepage
  \else
    \if@twocolumn\else\endquotation\fi
  \fi
}
```

Datei l
scrksect.dtx

12 Gliederung

Zur Gliederung gehören neben den üblichen Gliederungsbefehlen `\chapter` ... `\subparagraph` auch alle Optionen und Befehle, die diese Gliederung beeinflussen. Eingeschränkt gehören auch die Befehle dazu, die Inhaltsverzeichniseinträge zu den Gliederungsbefehlen erzeugen oder unmittelbar formatieren. Diese Entscheidung wurde getroffen, weil die Zusammengehörigkeit eher hier gesehen wird, als bei den Verzeichnissen selbst.

Briefe haben derzeit keine derartige Gliederung.

12.1 Optionen für die Gliederung

`open` Bei scrreprt und scrbook kann per Option eingestellt werden, ob Kapitel nur auf linken oder rechten Seiten beginnen dürfen oder auf jeder Seite.

`\if@openright` `\@openrighttrue` `\@openrightfalse` Der Zustand wird dann in einem Schalter und durch umdefinieren von `\cleardoublestandardpage` gespeichert.

```
\newif\if@openright
⟨report⟩\@openrightfalse
⟨book⟩\@openrighttrue
```

```
\KOMA@key{open}{%
  \KOMA@set@ncmdkey{open}{@tempa}{%
    {any}{0},%
    {right}{1},%
    {left}{2}%
  }{#1}%
  \ifcase \@tempa\relax
    \@openrightfalse
    \renewcommand*{\cleardoublestandardpage}{\cleardoubleoddstandardpage}%
  \or
    \@openrighttrue
    \renewcommand*{\cleardoublestandardpage}{\cleardoubleoddstandardpage}%
  \or
    \@openrighttrue
    \renewcommand*{\cleardoublestandardpage}{\cleardoubleevenstandardpage}%
  \fi
}
```

`openright` `openany`

```
\KOMA@DeclareStandardOption{openright}{open=right}
\KOMA@DeclareStandardOption{openany}{open=any}
```

numbers

Dies ist die zentrale Option zur Konfiguration der Nummerierung. Dabei geht es primär um die Gliederungsnummern, es sind jedoch auch davon abgeleitete Nummern betroffen.

scr@dotchangeatdocument

Da die Umschaltung zwischen automatischem Endpunkt, immer Endpunkt und kein Endpunkt unbedingt *vor* \begindocument erfolgen muss, wird eine Fehlermeldung ausgegeben, falls das jemand zu einem späteren Zeitpunkt probiert.

```
\newcommand*{\scr@dotchangeatdocument}[1]{%
  \ClassError{\KOMAClassName}{%
    change of end dot feature after \string\begin{document}%
  }{%
    You've tried to set 'numbers=#1' after \string\begin{document},\MessageBreak
    but this option is only allowed at the document preamble.\MessageBreak
    See KOMA-Script manual for more information about.%
  }%
}
```

Normalerweise wird automatisch entschieden, ob Nummern mit einem Punkt enden müssen oder nicht. Um dies abzuschalten, muss nur der gewünschte Zustand eingeschaltet und die Umschaltmöglichkeit entfernt werden. Die Umschaltung kann jedoch nicht direkt erfolgen, sondern erfolgt zeitverzögert in \begindocument. Hier wird nur das Makro umgeschaltet, das dort verwendet wird.

```
\KOMA@key{numbers}{%
  \KOMA@set@ncmdkey{numbers}{@tempa}{%
    {autoendperiod}{0},{autoenddot}{0},{auto}{0},%
    {endperiod}{1},{withendperiod}{1},{periodatend}{1},%
    {enddot}{1},{withenddot}{1},{dotatend}{1},%
    {noendperiod}{2},{noperiodatend}{2},%
    {noenddot}{2},{nodotatend}{2}%
  }{#1}%
  \ifcase \@tempa\relax
    \if@atdocument\scr@dotchangeatdocument{#1}\else
      \let\scr@altsecnumhook\@empty
    \fi
  \or
    \if@atdocument\scr@dotchangeatdocument{#1}\else
      \let\scr@altsecnumhook\scr@altsecnumhooktrue
    \fi
  \or
    \if@atdocument\scr@dotchangeatdocument{#1}\else
      \let\scr@altsecnumhook\scr@altsecnumhookfalse
    \fi
  \fi
}
```

\scr@altsecnumhook
\scr@altsecnumhooktrue
\scr@altsecnumhookfalse

Es werden drei Hilfsmakros benötigt. Das erste davon wird bei \begindocument ausgeführt (nachdem die aux-Datei gelesen wurde!) und schaltet den

gewünschten Zustand ein. Dazu wird es per Option ggf. auf die eine oder andere Bedeutung umgeschaltet. Im Auto-Fall ist es übrigens `\@empty`.

```
\newcommand*{\scr@altsecnumhook}{}
\AtBeginDocument{\scr@altsecnumhook}
\newcommand*{\scr@altsecnumhooktrue}{%
  \@altsecnumformattrue\global\let\@altsecnumformatfalse\@altsecnumformattrue
}
\newcommand*{\scr@altsecnumhookfalse}{%
  \@altsecnumformatfalse\global\let\@altsecnumformattrue\@altsecnumformatfalse
}
```

`pointednumbers`
`pointlessnumbers`

```
\KOMA@DeclareDeprecatedOption{pointednumbers}{numbers=enddot}
\KOMA@DeclareDeprecatedOption{pointlessnumbers}{numbers=noenddot}
```

Übrigens weiß ich, wie blödsinnig die Namen der Optionen sind. Aber das ist nun nicht mehr zu ändern.

`chapterprefix`
`nochapterprefix`

Normalerweise verwenden scrbook und scrreprt nicht die enorm großen Überschriften von book und report, die mit einem Absatz „Kapitel *Nummer*“ beginnen. Mit der Option `chapterprefix` kann dies jetzt wieder aktiviert werden. Die Option, zur Deaktivierung heißt entsprechend nochapterprefix.

`\if@chapterprefix`
`\@chapterprefixtrue`
`\@chapterprefixfalse`

Um für Spezialanwendungen auch innerhalb des Dokuments eine Umschaltung zu ermöglichen bzw. für die Optionen `appendixprefix` und `noappendixprefix` erfolgt die Umschaltung durch einen Schalter.

```
\newif\if@chapterprefix\@chapterprefixfalse
\KOMA@ifkey{chapterprefix}{@chapterprefix}
\KOMA@DeclareDeprecatedOption{nochapterprefix}{chapterprefix=false}
```

`appendixprefix`
`noappendixprefix`

Will man abweichend von der Option `chapterprefix` die großen Überschriften für den Anhang aktivieren oder deaktivieren, so kann man das mit dieser Option erreichen. Allerdings setzt diese keinen Schalter, sondern ein Zusatzmakro, das auch für andere Zwecke genutzt werden kann. Übrigens ist keine der beiden Einstellungen die Voreinstellung!

```
\KOMA@key{appendixprefix}[true]{%
  \KOMA@set@ifkey{appendixprefix}{@tempswa}{#1}%
  \if@tempswa
    \def\appendixmore{\@chapterprefixtrue}%
  \else
    \def\appendixmore{\@chapterprefixfalse}%
  \fi
}
\KOMA@DeclareDeprecatedOption{noappendixprefix}{appendixprefix=false}
```

\size@part
\size@partnumber
\size@chapter
\size@chapterprefix
\size@section
\size@subsection
\size@subsubsection
\size@paragraph
\size@subparagraph

Diese Makros werden intern von den Elementen benötigt (siehe unten). Darüber hinaus werden sie per Option geändert und sind deshalb vor der entsprechenden Option zu definieren.

```
\newcommand*{\size@part}{}
\newcommand*{\size@partnumber}{}
⟨book | report⟩\newcommand*{\size@chapter}{}
⟨book | report⟩\newcommand*{\size@chapterprefix}{\size@chapter}
\newcommand*{\size@section}{}
\newcommand*{\size@subsection}{}
\newcommand*{\size@subsubsection}{}
\newcommand*{\size@paragraph}{}
\newcommand*{\size@subparagraph}{}
```

\chapterheadstartvskip
\chapterheadendvskip

Mit diesen Anweisungen wird üblicherweise der Abstand zwischen oberem Rand des Textbereichs und der Überschrift bzw. der Abstand zwischen der Überschrift und dem Text des Kapitels gesetzt. Man kann die Anweisungen aber zusätzlich auch für andere Zwecke missbrauchen. Das ist der Grund warum diese Makros keine Längen, sondern komplette Anweisungen enthalten. In der Voreinstellung enthalten sie jedoch gar nichts. Die tatsächlichen Werte werden durch die Optionen festgelegt.

```
\newcommand*{\chapterheadstartvskip}{}
\newcommand*{\chapterheadendvskip}{}
```

headings

Dies ist eine neue zentrale Option für Überschriften. Dabei ergibt sich nun auch das Problem, dass die Gesamtmenge der Überschriften mal *headings* (Überschriften), mal *sectioning* (Abschnittseinteilung) und mal *disposition* (Gliederung) genannt wird. Es wäre gut gewesen, sich bei Zeiten Gedanken darüber zu machen, wie man das am besten einheitlich nennt.

```
\KOMA@key{headings}{%
  \KOMA@set@ncmdkey{headings}{@tempa}{%
    {big}{0},%
    {normal}{1},%
    {small}{2}%
⟨*book | report⟩
    ,{openany}{3},%
    {openright}{4},%
    {openleft}{5}%
    ,{twolinechapter}{6},{chapterprefix}{6},{chapterwithprefix}{6},%
    {chapterwithprefixline}{6},%
    {onelinechapter}{7},{nochapterprefix}{7},{chapterwithoutprefix}{7},%
    {chapterwithoutprefixline}{7}%
    ,{twolineappendix}{8},{appendixprefix}{8},{appendixwithprefix}{8},%
    {appendixwithprefixline}{8},%
    {onelineappendix}{9},{noappendixprefix}{9},{appendixwithoutprefix}{9},%
    {appendixwithoutprefixline}{9}%
⟨/book | report⟩
  }{#1}%
```

```
  \ifcase \@tempa\relax
```

Als erstes kümmern wir uns um die drei Größenabstufungen für die Überschriften, die KOMA-Script bietet, und zwar in der Reihenfolge `big`, `normal`, `small`. Es sei darauf hingewiesen, dass die Verwendung dieser Option alle Änderungen an `\chapterheadstartvskip` oder `\chapterheadendvskip` oder den Font-Elementen für die einzelnen Überschriften aufhebt macht!

```
⟨*book | report⟩
    \renewcommand*{\chapterheadstartvskip}{%
      \vspace*{2.3\baselineskip}%
    }%
    \renewcommand*{\chapterheadendvskip}{%
      \vspace{1.725\baselineskip
        \@plus .115\baselineskip \@minus .192\baselineskip}%
    }%
⟨/book | report⟩
    \renewcommand*{\size@part}{\Huge}%
    \renewcommand*{\size@partnumber}{\huge}%
⟨book | report⟩     \renewcommand*{\size@chapter}{\huge}%
    \renewcommand*{\size@section}{\Large}%
    \renewcommand*{\size@subsection}{\large}%
    \renewcommand*{\size@subsubsection}{\normalsize}%
    \renewcommand*{\size@paragraph}{\normalsize}%
    \renewcommand*{\size@subparagraph}{\normalsize}%
  \or
⟨*book | report⟩
    \renewcommand*{\chapterheadstartvskip}{%
      \vspace*{2\baselineskip}%
    }%
    \renewcommand*{\chapterheadendvskip}{%
      \vspace{1.5\baselineskip
        \@plus .1\baselineskip \@minus .167\baselineskip}%
    }%
⟨/book | report⟩
    \renewcommand*{\size@part}{\huge}%
    \renewcommand*{\size@partnumber}{\huge}%
⟨*book | report⟩
    \renewcommand*{\size@chapter}{\LARGE}%
    \renewcommand*{\size@section}{\Large}%
    \renewcommand*{\size@subsection}{\large}%
⟨/book | report⟩
⟨*article⟩
    \renewcommand*{\size@section}{\large}%
    \renewcommand*{\size@subsection}{\normalsize}%
⟨/article⟩
    \renewcommand*{\size@subsubsection}{\normalsize}%
    \renewcommand*{\size@paragraph}{\normalsize}%
    \renewcommand*{\size@subparagraph}{\normalsize}%
  \or
```

```
⟨*book | report⟩
    \renewcommand*{\chapterheadstartvskip}{%
      \vspace*{1.8\baselineskip}%
    }%
    \renewcommand*{\chapterheadendvskip}{%
      \vspace{1.35\baselineskip
        \@plus 0.09\baselineskip \@minus .15\baselineskip}%
    }%
⟨/book | report⟩
    \renewcommand*{\size@part}{\LARGE}%
    \renewcommand*{\size@partnumber}{\LARGE}%
⟨*book | report⟩
    \renewcommand*{\size@chapter}{\Large}%
    \renewcommand*{\size@section}{\large}%
⟨/book | report⟩
⟨*article⟩
    \renewcommand*{\size@section}{\normalsize}%
⟨/article⟩
    \renewcommand*{\size@subsection}{\normalsize}%
    \renewcommand*{\size@subsubsection}{\normalsize}%
    \renewcommand*{\size@paragraph}{\normalsize}%
    \renewcommand*{\size@subparagraph}{\normalsize}%
```

Für scrbook und scrreprt gibt es dann noch die Möglichkeit, auch hierüber die Optionen `open=any`, `open=right` und `open=left` zu setzen.

```
  \or
⟨*book | report⟩
    \KOMA@options{open=any}{}%
⟨/book | report⟩
  \or
⟨*book | report⟩
    \KOMA@options{open=right}{}%
⟨/book | report⟩
  \or
⟨*book | report⟩
    \KOMA@options{open=left}{}%
⟨/book | report⟩
```

Für scrbook und scrreprt gibt es dann noch die Möglichkeit, auch hierüber die Optionen `chapterprefix` und `appendixprefix` zu setzen.

```
  \or
⟨*book | report⟩
    \KOMA@options{chapterprefix=true}{}%
⟨/book | report⟩
  \or
⟨*book | report⟩
    \KOMA@options{chapterprefix=false}{}%
⟨/book | report⟩
  \or
⟨*book | report⟩
```

```
    \KOMA@options{appendixprefix=true}{}%
⟨/book | report⟩
  \or
⟨*book | report⟩
    \KOMA@options{appendixprefix=false}{}%
⟨/book | report⟩
  \fi
}
```

bigheadings
\@bigheadings
normalheadings
\@normalheadings
smallheadings
\@smallheadings

```
\KOMA@DeclareDeprecatedOption{bigheadings}{headings=big}
\KOMA@DeclareDeprecatedOption{normalheadings}{headings=normal}
\KOMA@DeclareDeprecatedOption{smallheadings}{headings=small}
```

12.2 Grobeinteilung von Büchern

\if@mainmatter
\@mainmattertrue
\@mainmatterfalse

Über diesen Schalter wird bei Buchklassen festgelegt, ob wir uns im Hauptteil des Dokuments befinden oder nicht. Die eigentliche Umschaltung geschieht mit Hilfe der Befehle `\frontmatter`, `\mainmatter`, `\backmatter`. Ohne Umschaltung befinden wir uns bereits im Hauptteil.

```
\newif\if@mainmatter\@mainmattertrue
```

\frontmatter

Der Vorspann enthält bei Büchern oftmals Inhaltsverzeichnis, Vorwort und dergleichen und wird mit kleinen römischen Seitenzahlen versehen.

```
\newcommand*\frontmatter{%
  \if@twoside\cleardoubleoddpage\else\clearpage\fi
  \@mainmatterfalse\pagenumbering{roman}%
}
```

\mainmatter

Der Hauptteil ist das Fleisch der Bücher.

```
\newcommand*\mainmatter{%
  \if@twoside\cleardoubleoddpage\else\clearpage\fi
  \@mainmattertrue\pagenumbering{arabic}%
}
```

\backmatter

Auch wenn nicht ganz einzusehen ist, warum das anders sein soll, wird hier nicht nach der Option `twoside`, sondern nach `openright` unterschieden.

```
\newcommand*\backmatter{%
  \if@openright\cleardoubleoddpage\else\clearpage\fi\@mainmatterfalse
}
```

12.3 Anhang

\appendix

Wird der Anhang aktiviert, so werden Kapitel zukünftig mit Buchstaben nummeriert.

\appendixmore

\appendixmore kann als *hook* verwendet werden.

```
\newcommand*\appendix{\par%
⟨*article⟩
  \setcounter{section}{0}%
  \setcounter{subsection}{0}%
  \gdef\thesection{\@Alph\c@section}%
⟨/article⟩
⟨*report | book⟩
  \setcounter{chapter}{0}%
  \setcounter{section}{0}%
  \gdef\@chapapp{\appendixname}%
  \gdef\thechapter{\@Alph\c@chapter}%
⟨/report | book⟩
  \csname appendixmore\endcsname
}
```

12.4 Definitionen für Seitenstile

\chaptermark

Da \chapter nicht mit Hilfe von \@startsection definiert wird, wird auch \chaptermark nicht automatisch definiert. Es wird aber in den Seitenstilen benötigt bzw. umdefiniert. Also hier vorsorglich eine Dummy-Definition:

```
⟨book | report⟩\newcommand*\chaptermark[1]{}
```

12.5 Der Automatismus für den Punkt in der Nummerierung

Nach Duden 20. Auflage, Regel 5 wird in der Gliederung kein Punkt gesetzt, wenn diese rein mit arabischen Zahlen erfolgt. Nach Regel 6 folgt jedoch ein Punkt, wenn die Gliederung auch römische Zahlen oder Großbuchstaben enthält. Aus Konsistenzgründen muss der Punkt in diesem Fall immer gesetzt werden. KOMA-Script bietet dafür einen Automatismus.

\if@altsecnumformat
\@altsecnumformattrue
\@altsecnumformatfalse

Dieses Schalter wird für die Umschaltung auf Duden-Regel 6 bei der Numerierung benötigt.

\if@autodot
\@autodottrue
\@autodotfalse

Damit die Umschaltung auch wieder zurück auf Regel 5 automatisch erfolgen kann, wird ein zweiter Schalter benötigt.

```
\newif\if@altsecnumformat\@altsecnumformatfalse
\newif\if@autodot
\renewcommand*{\@autodottrue}{\global\let\if@autodot\iftrue}
\renewcommand*{\@autodotfalse}{\global\let\if@autodot\iffalse}
\@autodotfalse
```

\@maybeautodot

Ob der automatische Punkt aktiviert werden muss, wird aufgrund der Darstellung eines Zählers entschieden. Diese wird dem Makro als Argument übergeben (\thepart, \thechapter etc.). Durch Verwendung einer Gruppe, werden alle Definitionen lokal gehalten.

```
\newcommand{\@maybeautodot}[1]{\if@autodot\else\begingroup%
```

```
  \expandafter\@@maybeautodot #1\@stop\endgroup\fi
}
```

\@@maybeautodot Die expandierte Darstellung wird dann auf Darstellungsbefehle für Zähler gescannt.

```
\newcommand*{\@@maybeautodot}[1]{%
  \ifx #1\@stop\let\@@maybeautodot\relax
  \else
    \ifx #1\Alph \@autodottrue\fi
    \ifx #1\alph \@autodottrue\fi
    \ifx #1\Roman \@autodottrue\fi
    \ifx #1\roman \@autodottrue\fi
    \ifx #1\@Alph \@autodottrue\fi
    \ifx #1\@alph \@autodottrue\fi
    \ifx #1\@Roman \@autodottrue\fi
    \ifx #1\@roman \@autodottrue\fi
    \ifx #1\romannumeral \@autodottrue\fi
  \fi
  \@@maybeautodot
}
```

\scr@sect \@sect Damit auch alle mit \@startsection definierten Gliederungsebenen bei der Entscheidung berücksichtigt werden, muss ein internes Makro des LaTeX-Kerns erweitert werden. Für die Ebenen \part und \chapter wird der Test direkt in der Definition der Befehle erledigt.

```
\let\scr@sect\@sect
\renewcommand*{\@sect}[1]{\stepcounter{#1}%
  \expandafter\@maybeautodot\csname the#1\endcsname
  \addtocounter{#1}{-1}\scr@sect{#1}%
}
```

Dabei soll die Umstellung beim zweiten TeX-Lauf bereits am Anfang aktiv sein. Deshalb wird die Information am Ende in die aux-Datei geschrieben.

```
\BeforeClosingMainAux{%
  \if@autodot\if@filesw\immediate\write\@mainaux{%
      \string\global\string\@altsecnumformattrue}%
  \fi\fi}
```

\autodot Dieses Makro setzt dann den Punkt nach Bedarf.

```
\newcommand*\autodot{\if@altsecnumformat.\fi}
```

12.6 Zähler für die Gliederungsebenen

secnumdepth Dieser Zähler gibt an, bis zu welcher Ebene die Gliederungsüberschriften nummeriert werden.

```
⟨book | report⟩\setcounter{secnumdepth}{2}
⟨article⟩\setcounter{secnumdepth}{3}
```

part
\thepart
\partformat
chapter
\thechapter
\chapterformat
\chaptermarkformat
section
\thesection
\sectionmarkformat
subsection
\thesubsection
\subsectionmarkformat
subsubsection
\thesubsubsection
paragraph
\theparagraph
subparagraph
\thesubparagraph

Jede Gliederungsebene benötigt einen Zähler für die Gliederungsnummer und eine Darstellung des Zählers (`\the...`). Desweiteren wird eine Formatierung des Zählers in den Gliederungsüberschriften (`\...format`) und eine Formatierung des Zählers in der Kopfzeile (`\...markformat`) benötigt.

```
\newcounter{part}
\renewcommand*{\thepart}{\@Roman\c@part}
\newcommand*{\partformat}{\partname~\thepart\autodot}
⟨*book | report⟩
\newcounter{chapter}
\renewcommand*{\thechapter}{\@arabic\c@chapter}
\newcommand*{\chapterformat}{%
  \mbox{\chapappifchapterprefix{\nobreakspace}\thechapter\autodot\enskip}%
}
\newcommand*\chaptermarkformat{\chapappifchapterprefix{\ }%
  \thechapter\autodot\enskip}
\newcounter{section}[chapter]
\renewcommand*\thesection{%
⟨*book⟩
  \ifnum \scr@compatibility>\@nameuse{scr@v@2.97d} % the space before this
                                % comment should be \relax but hyperref has a
                                % not documented problem with \relax
    \if@mainmatter\thechapter.\fi
  \else
⟨/book⟩
  \thechapter.%
⟨book⟩  \fi
  \@arabic\c@section
}
\newcommand*\sectionmarkformat{\thesection\autodot\enskip}
⟨/book | report⟩
⟨*article⟩
\newcounter{section}
\renewcommand*{\thesection}{\@arabic\c@section}
\newcommand*\sectionmarkformat{\thesection\autodot\enskip}
⟨/article⟩
\newcounter{subsection}[section]
\renewcommand*{\thesubsection}{\thesection.\@arabic\c@subsection}
⟨*article⟩
\newcommand*\subsectionmarkformat{\thesubsection\autodot\enskip}
⟨/article⟩
\newcounter{subsubsection}[subsection]
\renewcommand*{\thesubsubsection}{%
  \thesubsection.\@arabic\c@subsubsection
}
\newcounter{paragraph}[subsubsection]
\renewcommand*{\theparagraph}{\thesubsubsection.\@arabic\c@paragraph}
\newcounter{subparagraph}[paragraph]
\renewcommand*{\thesubparagraph}{%
  \theparagraph.\@arabic\c@subparagraph
```

```
}
```

\@seccntformat
\othersectionlevelsformat

Bei den Ebenen ab \section erfolgt die Formatierung der Gliederungnummer in der Überschrift mit einer gemeinsamen Anweisung. Die interne Anweisung \@seccntformat wird dabei von LaTeX-Kern verwendet und muss umdefiniert werden, um eine entsprechende Anweisung auf Anwenderebene verfügbar zu haben. Ab Version 2.9p wurde in \@seccntformat ein \protect vor \othersectionlevelsformat eingefügt. Damit wurde es für den Benutzer deutlich einfacher, die Anweisung \othersectionlevelsformat umzudefinieren. Dafür war dadurch aber \the... nicht mehr expandiert. Das wiederum gab Probleme, wenn \seccntformat für den \mark-Mechanismus genutzt wurde. Deshalb hat \othersectionlevelsformat ab Version 2.97 drei Argumente statt nur eines. Das zweite Argument sollte vom Anwender ignoriert werden, es existiert nur, um die Kompatibitlität Umdefinitionen zu erhalten, bei denen \othersectionlevelsformat nur ein Argument besitzt. Das dritte Argument ist die \the...-Anweisung. Das zweite Argument funktioniert so, das es das dritte Argument frisst, falls \othersectionlevelsformat nur ein Argument liest.

```
\newcommand*{\othersectionlevelsformat}[3]{#3\autodot\enskip}
\renewcommand*{\@seccntformat}[1]{%
   \protect\othersectionlevelsformat{#1}{%
     \expandafter\aftergroup\noexpand\@gobble}{\csname the#1\endcsname}%
}
```

12.7 Namen für die Gliederung

\partname
\chaptername
\abstractname

Diese Makros enthalten die Namen der Gliederungsebenen \part und \chapter, sowie den Namen der Zusammenfassung:

```
\newcommand*\partname{Part}
⟨book | report⟩\newcommand*\chaptername{Chapter}
⟨report | article⟩\newcommand*\abstractname{Abstract}
```

\appendixname

Der Name des Anhangs.

```
\newcommand*\appendixname{Appendix}
```

\chapappifprefix
\chapappifchapterprefix
\chapapp

Das in \chapterformat verwendete Makro \chapappifchapterprefix setzt abhängig von \if@chapterprefix noch das Makro \chapapp gefolgt vom obliatorischen Argument. \chapapp macht \@chapapp auf Anwenderebene verfügbar. Der Vorteil des neuen \chapappifchapterprefix gegenüber dem alten \chapappifprefix ist, dass nun beim Umdefinieren von \chapterformat und \chaptermarkformat wieder \renewcommand verwendet werden kann und beim Umschalten in den Anhang die Kopfzeile korrekt ist, weil das Makro nun nicht mehr geschützt ist, sondern direkt expandiert.

```
\newcommand*{\chapappifprefix}[1][]{%
  \ClassWarning{\KOMAClassName}{%
    Please don't use obsolete command
    \string\chapappifprefix.\MessageBreak
    The new command \string\chapappifchapterprefix\space has
    an\MessageBreak
    obligatory instead of an optional argument. Use that\MessageBreak
    new command%
  }%
  \if@chapterprefix\chapapp#1\fi
}
\newcommand*{\chapappifchapterprefix}[1]{%
  \if@chapterprefix\chapapp#1\fi
}
\newcommand*{\chapapp}{\@chapapp}
```

\@chapapp Dieses Makro enthält den Kapitelname (\chaptername), der sich im Anhang üblicherweise ändert (\appendixname).

```
\newcommand*\@chapapp{\chaptername}
```

12.8 Die Gliederungsbefehle

\part
\@part
\@spart
\@endpart

Der Gliederungsfehl \part wird wie in den Standardklassen mit Hilfe mehrere Hilfsmakros definiert. Wesentliche Unterschiede zu den Standardklassen existieren nicht, jedoch sind Schriftgröße, Schriftart u. ä. flexibler gehalten. Außerdem gibt es die Möglichkeit von Präambeln. *Achtung: Die Präambel vor der \part-Überschrift kann ohne Warnung über den oberen Rand hinaus reichen!*

\partheadstartvskip
\partheadmidvskip
\partheadendvskip
\parheademptypage

Diese vier Makros werden verwendet, um den Abstand über in und unter der Überschrift eines Teils und die leere Rückseite zu definieren. Prinzipiell kann damit auch der Umbruch zwischen Nummer und Text entfernt werden.

```
\newcommand*{\partheadstartvskip}{%
⟨article⟩  \addvspace{4ex}%
⟨report | book⟩   \null\vfil
}
\newcommand*{\partheadmidvskip}{%
  \par\nobreak
⟨book | report⟩   \vskip 20\p@
}
\newcommand*{\partheadendvskip}{%
⟨article⟩      \vskip 3ex
⟨report | book⟩   \vfil\newpage
}
⟨*report | book⟩
\newcommand*{\partheademptypage}{%
  \if@twoside\if@openright
      \null%
```

```
      \thispagestyle{empty}%
      \newpage
  \fi\fi
}
⟨/report | book⟩
⟨*article⟩
\newcommand\part{\par
  \partheadstartvskip%
  \@afterindentfalse
  \secdef\@part\@spart
}
⟨/article⟩
⟨*report | book⟩
\newcommand\part{\if@openright\cleardoublepage\else\clearpage\fi
  \thispagestyle{\partpagestyle}%
  \if@twocolumn
    \onecolumn
    \@tempswatrue
  \else
    \@tempswafalse
  \fi
  \partheadstartvskip
  \vbox to\z@{\vss\use@preamble{part@o}\strut\par}%
  \vskip-\baselineskip\nobreak%
  \secdef\@part\@spart
}
⟨/report | book⟩
\newcommand*{\@part}{}
\def\@part[#1]#2{%
⟨article⟩  \ifnum \c@secnumdepth >\m@ne
⟨report | book⟩  \ifnum \c@secnumdepth >-2\relax
    \refstepcounter{part}%
    \@maybeautodot\thepart%
    \addparttocentry{\thepart}{#1}%
  \else
    \addparttocentry{}{#1}%
  \fi
  \begingroup
    \setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
    \raggedpart
    \interlinepenalty \@M
    \normalfont\sectfont\nobreak
⟨article⟩    \ifnum \c@secnumdepth >\m@ne
⟨book | report⟩    \ifnum \c@secnumdepth >-2\relax
      \size@partnumber{\partformat}%
      \partheadmidvskip
    \fi
    \size@part{#2}\strut%
    \partmark{#1}\par
```

```
  \endgroup
⟨*article⟩
  \nobreak
  \partheadendvskip
  \@afterheading
⟨/article⟩
⟨book | report⟩   \@endpart
}
\newcommand*{\@spart}[1]{%
  \begingroup
    \setparsizes{\z@}{\z@}{\z@\@plus 1fil}\par@updaterelative
    \raggedpart
    \interlinepenalty \@M
    \normalfont
    \sectfont\nobreak\size@part{#1}\strut\@mkboth{}{}\par
  \endgroup
⟨*article⟩
  \nobreak
  \partheadendvskip
  \@afterheading
⟨/article⟩
⟨book | report⟩   \@endpart
}
⟨*book | report⟩
\newcommand*{\@endpart}{\vbox to\z@{\use@preamble{part@u}\vss}%
  \partheadendvskip
  \partheademptypage
  \if@tempswa
    \twocolumn
  \fi
}
⟨/book | report⟩
```

`\addparttocentry` Seit Version 3.08 wird der Eintrag nicht direkt innerhalb von `\part` per `\addcontentsline` erzeugt, sondern indirekt über diese Anweisung. Das erste Argument ist dabei die (formatierte) Nummer bzw. bei nicht nummerierten Teilen leer. Das zweite Argument ist der Überschriftstext für das Verzeichnis. Durch diesen indirekten Weg, kann die Anweisung einfach umdefiniert werden. Verwendet wird hier die Standardanweisung für Inhaltsverzeichniseinträge, die in `scrkliof.dtx` definiert ist:

```
\newcommand*{\addparttocentry}[2]{%
  \addtocentrydefault{part}{#1}{#2}%
}
```

`\addpart`
`\@addpart`
`\@saddpart`

```
\newcommand\addpart{%
⟨*article⟩
  \par
```

```
  \partheadstartvskip%
  \@afterindentfalse
⟨/article⟩
⟨*report | book⟩
  \if@openright\cleardoublepage\else\clearpage\fi
  \thispagestyle{\partpagestyle}%
  \if@twocolumn
    \onecolumn
    \@tempswatrue
  \else
    \@tempswafalse
  \fi
  \partheadstartvskip
  \vbox to\z@{\vss\use@preamble{part@o}\strut\par}%
⟨/report | book⟩
  \secdef\@addpart\@saddpart
}
\newcommand*{\@addpart}{}
\def\@addpart[#1]#2{%
```

Notlösung, damit hyperref halbwegs damit klar kommt. Das ist aber weit davon entfernt eine echte Lösung zu bringen.

```
  \csname phantomsection\endcsname
  \addparttocentry{}{#1}%
  \@spart{#2}%
  \ifx\partmark\@gobble\else\@mkboth{#1}{}\fi
}
\newcommand*{\@saddpart}{\@spart}
```

\partmark

```
\newcommand*{\partmark}{}
\let\partmark\@gobble
```

\chapter
\@chapter
\@makechapterhead
\@@makechapterhead
\if@at@twocolumn
\scr@topnewpage
\@topnewpage

Der Gliederungsfehl `\chapter`, der nur in den Klassen scrbook und scrreprt vorhanden ist, wird wie in den Standardklassen mit Hilfe mehrere Hilfsmakros definiert.

Interessant ist dabei die Erweiterung, daß jedes Kapitel mit einer Präambel versehen werden kann. Diese wird im zweispaltigen Satz wie die Überschrift selbst einspaltig gesetzt. Intern wird dies für die Preambel des Index und der Bibliography verwendet. Dabei entsteht aber wiederum das Problem, dass der Indexkopf innerhalb von `\twocolumn` ausgegeben wird andererseits aber innerhalb von `\chapter` das Makro `\@topnewpage` verwendet wird. Damit würde dann also `\@topnewpage` innerhalb von `\@topnewpage` aufgerufen. Das ist jedoch nicht erlaubt. Als muss der Aufruf von `\@topnewpage` in `\@chapter` und `\@schapter` verhindert werden, wenn wir uns bereits in `\@topnewpage` befinden. Eine Möglichkeit dafür wäre, einen Schalter in den Index-Kopf einzubauen. Ich wähle hingegen die Methode,

\@topnewpage entsprechend umzudefinieren. Damit kann dann auch der Anwender \@chapter innerhalb des optionalen Arguments von \twocolumn verwenden. Ein Schalter wird allerdings trotzdem gebraucht.

```
\newif\if@at@twocolumn
\newcommand*\scr@topnewpage{}
\let\scr@topnewpage\@topnewpage
\long\def\@topnewpage[#1]{%
  \@at@twocolumntrue\scr@topnewpage[{#1}]\@at@twocolumnfalse
}
```

\use@chapter@o@preamble

Das Makro wird benötigt, weil das Setzen der oberen Präambel etwas mehr Aufwand bedeutet. Die Anweisung spart Platz, da sie zweimal verwendet wird.

```
\newcommand*{\use@chapter@o@preamble}{%
  {\settoheight{\@tempdima}{\vbox{\chapterheadstartvskip}}%
    \settodepth{\@tempdimb}{\vbox{\chapterheadstartvskip}}%
    \addtolength{\@tempdima}{\@tempdimb}%
    \setbox0\vbox{\use@preamble{chapter@o}%
      \vskip-\parskip}%
    \setlength{\@tempdimb}{\ht0}%
    \addtolength{\@tempdimb}{\dp0}%
    \vbox to \z@{%
      \vbox to \@tempdima{\vfill\box0}%
      \vskip-\@tempdima%
      }%
    \ifdim \@tempdimb>\@tempdima%
      \addtolength{\@tempdimb}{-\@tempdima}%
      \ClassWarning{\KOMAClassName}{%
        preamble before chapter is \the\@tempdimb\space too
        high.\MessageBreak
        To avoid the overfull \string\vbox\space you may
        redefine\MessageBreak
        \string\chapterheadstartvskip\space at the preamble of
        your\MessageBreak
        document. You may also change
        \string\setchapterpreamble\MessageBreak
        before the command, which generates the message
        about\MessageBreak
        an overfull \string\vbox
        }%
    \fi%
    }
}
```

Zurück zur Kapiteldefinition:

```
\newcommand\chapter{\if@openright\cleardoublepage\else\clearpage\fi
  \thispagestyle{\chapterpagestyle}%
  \global\@topnum\z@
```

```
  \@afterindentfalse
  \secdef\@chapter\@schapter
}
```

Die Variante ohne Stern:

```
\newcommand*{\@chapter}{}
\def\@chapter[#1]#2{\ifnum \c@secnumdepth >\m@ne
⟨book⟩     \if@mainmatter
      \refstepcounter{chapter}%
      \@maybeautodot\thechapter
      \typeout{\@chapapp\space\thechapter.}%
      \addchaptertocentry{\thechapter}{#1}%
      \if@chaptertolists
        \doforeachtocfile{%
          \iftocfeature{\@currext}{chapteratlist}{%
            \addcontentsline{\@currext}{chapteratlist}{%
              \protect\numberline{\thechapter}#1}%
          }{}%
        }%
        \@ifundefined{float@addtolists}{}{%
          \scr@float@addtolists@warning
          \scr@ifundefinedorrelax{@currentHref}{%
            \float@addtolists{%
              \protect\contentsline{chapteratlist}{%
                \protect\numberline{\thechapter}#1}{\thepage}%
            }%
          }{%
            \float@addtolists{%
              \protect\contentsline{chapteratlist}{%
                \protect\numberline{\thechapter}#1}{\thepage}{\@currentHref}%
            }%
          }%
        }%
      \fi
⟨*book⟩
    \else
      \addchaptertocentry{}{#1}%
      \if@chaptertolists
        \doforeachtocfile{%
          \iftocfeature{\@currext}{chapteratlist}{%
            \addcontentsline{\@currext}{chapteratlist}{#1}%
          }{}%
        }%
        \@ifundefined{float@addtolists}{}{%
          \scr@float@addtolists@warning
          \scr@ifundefinedorrelax{@currentHref}{%
            \float@addtolists{%
              \protect\contentsline{chapteratlist}{#1}{\thepage}%
            }%
          }{%
```

```
            \float@addtolists{%
              \protect\contentsline{chapteratlist}{#1}{\thepage}{\@currentHref}%
            }%
          }%
        }%
      \fi
    \fi
⟨/book⟩
  \else
    \addchaptertocentry{}{#1}%
    \if@chaptertolists
      \doforeachtocfile{%
        \iftocfeature{\@currext}{chapteratlist}{%
          \addcontentsline{\@currext}{chapteratlist}{#1}%
        }{}%
      }%
      \@ifundefined{float@addtolists}{}{%
        \scr@float@addtolists@warning
        \scr@ifundefinedorrelax{@currentHref}{%
          \float@addtolists{%
            \protect\contentsline{chapteratlist}{#1}{\thepage}%
          }%
        }{%
          \float@addtolists{%
            \protect\contentsline{chapteratlist}{#1}{\thepage}{\@currentHref}%
          }%
        }%
      }%
    \fi
  \fi
  \chaptermark{#1}%
  \ifdim \@chapterlistsgap>\z@
    \doforeachtocfile{%
      \iftocfeature{\@currext}{chapteratlist}{%
        \addtocontents{\@currext}{\protect\addvspace{\@chapterlistsgap}}%
      }{}%
    }%
    \@ifundefined{float@addtolists}{}{%
      \scr@float@addtolists@warning
      \float@addtolists{\protect\addvspace{\@chapterlistsgap}}%
    }%
  \fi
  \if@twocolumn
    \if@at@twocolumn
      \@makechapterhead{#2}%
    \else
      \@topnewpage[\@makechapterhead{#2}]%
    \fi
  \else
```

```
    \@makechapterhead{#2}%
    \@afterheading
  \fi
}
\newcommand*{\@makechapterhead}[1]{%
  \use@chapter@o@preamble
  \@@makechapterhead{#1}%
  \use@preamble{chapter@u}\nobreak
}
```

```
\newcommand*{\@@makechapterhead}[1]{\chapterheadstartvskip
  {%
    \setlength{\parindent}{\z@}\setlength{\parfillskip}{\fill}%
    \normalfont\sectfont\nobreak\size@chapter{}%
    \if@chapterprefix
      \let\@tempa\raggedsection
    \else
      \let\@tempa\@hangfrom
    \fi
    \@tempa{\ifnum \c@secnumdepth >\m@ne%
⟨book⟩        \if@mainmatter
          \if@chapterprefix
            \expandafter\size@chapterprefix
          \else
            \expandafter\size@chapter
          \fi
          {\chapterformat}%
          \if@chapterprefix
            \size@chapterprefix{}\endgraf\nobreak\vskip.5\baselineskip
          \fi
⟨book⟩        \fi
      \fi
    }%
    {\raggedsection \interlinepenalty \@M \size@chapter{#1}\par}}%
  \nobreak\chapterheadendvskip
}
```

`\addchaptertocentry` Seit Version 3.08 wird der Eintrag nicht direkt innerhalb von `\chapter` per `\addcontentsline` erzeugt, sondern indirekt über diese Anweisung. Das erste Argument ist dabei die (formatierte) Nummer bzw. bei nicht nummerierten Kapiteln leer. Das zweite Argument ist der Überschriftstext für das Verzeichnis. Durch diesen indirekten Weg, kann die Anweisung einfach umdefiniert werden. Verwendet wird hier die Standardanweisung für Inhaltsverzeichniseinträge, die in `scrkliof.dtx` definiert ist:

```
\newcommand*{\addchaptertocentry}[2]{%
  \addtocentrydefault{chapter}{#1}{#2}%
}
```

`\@schapter` `\@makeschapterhead` `\@@makeschapterhead` Die Sternvariante, die um Einiges einfacher ausfällt.

```
\newcommand*{\@schapter}[1]{%
  \if@twocolumn
    \if@at@twocolumn
      \@makeschapterhead{#1}%
    \else
      \@topnewpage[\@makeschapterhead{#1}]%
    \fi
  \else
    \@makeschapterhead{#1}%
    \@afterheading
  \fi
}
\newcommand*{\@@makeschapterhead}[1]{%
  \chapterheadstartvskip%
  {\normalfont\sectfont\nobreak\size@chapter{}%
    \setlength{\parindent}{\z@}\setlength{\parfillskip}{\fill}%
    \raggedsection \interlinepenalty \@M \size@chapter{#1}\par}%
  \nobreak\chapterheadendvskip%
}
\newcommand*{\@makeschapterhead}[1]{%
  \use@chapter@o@preamble
  \@@makeschapterhead{#1}%
  \use@preamble{chapter@u}\nobreak
}
```

`\addchap` KOMA-Script bietet in scrbook und scrreprt den zusätzlichen Gliederungsbefehl `\addchap`. Es handelt sich dabei um einen mit `\chapter*` vergleichbaren Befehl, bei dem jedoch der Kolumnentitel korrigiert wird und ein Eintrag ins Inhaltsverzeichnis erfolgt (nicht in der Stern-Variante).

Die Definition von `\addcap` sah bis zu Version 2.7 so aus:

```
\newcommand\addchap{\if@openright\cleardoublepage\else\clearpage\fi
                    \thispagestyle{plain}%
                    \global\@topnum\z@
                    \@afterindentfalse
                    \secdef\@addchap\@saddchap}
\def\@addchap[#1]#2{\typeout{#2}
                    \addcontentsline{toc}{chapter}{#1}
                    \if@twoside\@mkboth{#1}{}\else\@mkboth{}{#1}\fi
                    \addtocontents{lof}{\protect\addvspace{10\p@}}%
                    \addtocontents{lot}{\protect\addvspace{10\p@}}%
                    \if@twocolumn
                         \@topnewpage[\@makeschapterhead{#2}]%
                    \else
                         \@makeschapterhead{#2}%
                         \@afterheading
                    \fi}
\def\@saddchap#1{\@mkboth{}{}
                 \if@twocolumn
                      \@topnewpage[\@makeschapterhead{#1}]%
```

```
\else
     \@makeschapterhead{#1}%
     \@afterheading
\fi}
```

Dies führte jedoch zu Problemen im Zusammenhang mit hyperref, da hierbei keine korrekten Links erzeugt werden. Mit dem `hypertex`-Treiber von hyperref bis Version 6.71a werden auch in der neuen Implementierung keine korrekten Links erzeugt. Dies ist jedoch ein Bug in jenem Treiber. Mit neuen Versionen tritt das Problem nicht mehr auf. Deshalb wurde der Code wie folgt vereinfacht. Von wesentlicher Bedeutung ist dabei der Aufruf von `\chapter*`, da in `hpdftex.def` das interne Makro `\@schapter` so umdefiniert wird, dass ein korrekter Link erzeugt wird.

`\@addchap`
`\@saddchap`

```
\newcommand\addchap{\if@openright\cleardoublepage\else\clearpage\fi
  \thispagestyle{\chapterpagestyle}%
  \global\@topnum\z@
  \@afterindentfalse
  \secdef\@addchap\@saddchap%
}%
\newcommand*{\@addchap}{}%
\def\@addchap[#1]#2{\@schapter{#2}%
  \addchaptertocentry{}{#1}%
  \if@chaptertolists
    \doforeachtocfile{%
      \iftocfeature{\@currext}{chapteratlist}{%
        \addcontentsline{\@currext}{chapteratlist}{#1}%
      }{}%
    }%
    \@ifundefined{float@addtolists}{}{%
      \scr@float@addtolists@warning
      \scr@ifundefinedorrelax{@currentHref}{%
        \float@addtolists{%
          \protect\contentsline{chapteratlist}{#1}{\thepage}%
        }%
      }{%
        \float@addtolists{%
          \protect\contentsline{chapteratlist}{#1}{\thepage}{\@currentHref}%
        }%
      }%
    }%
  \fi
  \ifnum \@nameuse{scr@v@2.96}<\scr@compatibility\relax
    \ifdim \@chapterlistsgap>\z@
      \doforeachtocfile{%
        \iftocfeature{\@currext}{chapteratlist}{%
          \addtocontents{\@currext}{\protect\addvspace{\@chapterlistsgap}}%
        }{}%
```

```
      }%
      \@ifundefined{float@addtolists}{}{%
        \scr@float@addtolists@warning
        \float@addtolists{\protect\addvspace{\@chapterlistsgap}}%
      }%
    \fi
  \fi
  \if@twoside\@mkboth{#1}{}\else\@mkboth{}{#1}\fi
}%
\newcommand*{\@saddchap}[1]{\@schapter{#1}\@mkboth{}{}}%
```

In `hyperref` bis Version 6.71a ist ein dicker Bug. In verschiedenen `def`-Dateien wird dort das *interne* KOMA-Script-Makro `\@addchap` umdefiniert, ohne sicherzustellen, dass das Makro auch noch so aussieht, wie man das erwartet hat. Die sauberste Lösung wäre sicher gewesen dort `\@addchap` nur zu ergänzen, so wie man das auch bei `\@schapter` gemacht hatte. Ab Version 6.71b ist dieser Bug behoben. Damit ich nicht eines Fehlers verdächtigt werde, wird ggf. eine entsprechende Meldung ausgegeben.

```
\AfterPackage{hyperref}{%
  \@ifpackagelater{hyperref}{2001/02/19}{}{%
    \ClassWarningNoLine{\KOMAClassName}{%
      You are using an old version of hyperref package!\MessageBreak%
      This version has a buggy hack at many drivers\MessageBreak%
      causing \string\addchap\space to behave strange.\MessageBreak%
      Please update hyperref to at least version
      6.71b}}}
```

`\addsec` `\@addsec` `\@saddsec`

KOMA-Script bietet in allen drei Hauptklassen den zusätzlichen Gliederungsbefehl `\addsec`. Es handelt sich dabei um einen mit `\section*` vergleichbaren Befehl, bei dem jedoch der Kolumnentitel korrigiert wird und ein Eintrag ins Inhaltsverzeichnis erfolgt (nicht in der Stern-Variante).

```
\newcommand*\addsec{\secdef\@addsec\@saddsec}
\newcommand*{\@addsec}{}
\def\@addsec[#1]#2{\section*{#2}\addcontentsline{toc}{section}{#1}
⟨article⟩   \if@twoside \@mkboth{#1}{}\else \@mkboth{#1}{#1}\fi
⟨report | book⟩   \if@twoside\ifx\@mkboth\markboth\markright{#1}\fi\fi
}
\newcommand*{\@saddsec}[1]{\section*{#1}\@mkboth{}{}}
```

TODO: Was für `\part` und `\chapter` mit `\add...tocentry` bereits funktioniert, geht für die unteren Ebenen noch nicht, weil es dazu notwendig ist `\@startsection` umzudefinieren. Vorgesehen ist das, sobald diese Anweisung aus irgendwelchen Gründen ohnehin umdefiniert wird.

`\section` `\subsection` `\subsubsection` `\paragraph` `\subparagraph`

Die Standardgliederungsbefehle `\section` bis `\subparagraph` sind mit der im Kernel dafür vorgesehenen Schnittstelle definiert. Im Unterschied zu den Standardklassen erfolgt die Formatierung jedoch in `\raggedsection` und die Ausgabe in der Schriftart `\sectfont`. Außerdem sind die Größen

veränderbar. In letzter Konsequenz sollten hier eigentlich auch die Abstände variabilisiert werden. Vielleicht in einer späteren Version.

```
\newcommand\section{\@startsection{section}{1}{\z@}%
  {-3.5ex \@plus -1ex \@minus -.2ex}%
  {2.3ex \@plus.2ex}%
  {\ifnum \scr@compatibility>\@nameuse{scr@v@2.96}\relax
    \setlength{\parfillskip}{\z@ plus 1fil}\fi
    \raggedsection\normalfont\sectfont\nobreak\size@section}%
}
\newcommand\subsection{\@startsection{subsection}{2}{\z@}%
  {-3.25ex\@plus -1ex \@minus -.2ex}%
  {1.5ex \@plus .2ex}%
  {\ifnum \scr@compatibility>\@nameuse{scr@v@2.96}\relax
    \setlength{\parfillskip}{\z@ plus 1fil}\fi
    \raggedsection\normalfont\sectfont\nobreak\size@subsection
  }%
}
\newcommand\subsubsection{\@startsection{subsubsection}{3}{\z@}%
  {-3.25ex\@plus -1ex \@minus -.2ex}%
  {1.5ex \@plus .2ex}%
  {\ifnum \scr@compatibility>\@nameuse{scr@v@2.96}\relax
    \setlength{\parfillskip}{\z@ plus 1fil}\fi
    \raggedsection\normalfont\sectfont\nobreak\size@subsubsection
  }%
}
\newcommand\paragraph{\@startsection{paragraph}{4}{\z@}%
  {3.25ex \@plus1ex \@minus.2ex}%
  {-1em}%
  {\raggedsection\normalfont\sectfont\nobreak\size@paragraph}%
}
\newcommand\subparagraph{\@startsection{subparagraph}{5}{\parindent}%
  {3.25ex \@plus1ex \@minus .2ex}%
  {-1em}%
  {\raggedsection\normalfont\sectfont\nobreak\size@subparagraph}%
}
```

`\minisec`
`minisec`

In KOMA-Script gibt es diese zusätzliche Gliederungsebene, die immer ohne Nummer und ohne Eintrag ins Inhaltsverzeichnis erfolgt.

```
\newkomafont{minisec}{}
\newcommand\minisec[1]{\@afterindentfalse \vskip 1.5ex
  {\parindent \z@
    \ifnum \scr@compatibility>\@nameuse{scr@v@2.96}\relax
      \setlength{\parfillskip}{\z@ plus 1fil}\fi
    \raggedsection\normalfont\sectfont\nobreak
    \usekomafont{minisec}#1\par\nobreak}\nobreak%
  \@afterheading
}
```

12.9 Schrift und Formatierung

`\raggedsection` Dieses Makro gibt an, wie die Überschriften formatiert werden. Voreingestellt ist linksbündig.

```
\newcommand*{\raggedsection}{}
\let\raggedsection\raggedright
```

`\raggedpart` Die Ausrichtung der Überschrift von `\part` kann davon abweichend über `\raggedpart` konfiguriert werden. Die Voreinstellung ist von der Klasse abhängig.

```
\newcommand*{\raggedpart}{}
⟨article⟩\let\raggedpart\raggedsection
⟨report | book⟩\let\raggedpart\centering
```

`\sectfont` Dies Schriftart nicht nur für die Gliederungsbefehle. Dieses Makro ist als intern zu betrachten. Der Anwender sollte stattdessen das entsprtechende Fontelement verwenden.

```
⟨/class⟩
\newcommand*\sectfont{\normalcolor\sffamily\bfseries}
⟨*class⟩
```

`partentry` Schrift für den `\part`-Eintrag im Inhaltsverzeichnis.

```
\newkomafont{partentry}{\usekomafont{disposition}\large}
```

`partentrypagenumber` Schrift für die Seitenzahl des `\part`-Eintrags im Inhaltsverzeichnis abweichend von `partentry`.

```
\newkomafont{partentrypagenumber}{}
```

`chapterentry` Schrift für den `\chapter`-Eintrag im Inhaltsverzeichnis.

```
⟨*book | report⟩
\newkomafont{chapterentry}{\usekomafont{disposition}}
```

`chapterentrypagenumber` Schrift für die Seitenzahl des `\chapter`-Eintrags im Inhaltsverzeichnis abweichend von `chapterentry`.

```
\newkomafont{chapterentrypagenumber}{}
⟨/book | report⟩
```

`sectionentry` Schrift für den `\section`-Eintrag im Inhaltsverzeichnis.

```
⟨*article⟩
\newkomafont{sectionentry}{\usekomafont{disposition}}
```

`sectionentrypagenumber` Schrift für die Seitenzahl des `\section`-Eintrags im Inhaltsverzeichnis abweichend von `sectionentry`.

```
\newkomafont{sectionentrypagenumber}{}
⟨/article⟩
```

\scr@fnt@disposition
\scr@fnt@sectioning
\scr@fnt@part
\scr@fnt@partnumber
\scr@fnt@chapter
\scr@fnt@chapterprefix
\scr@fnt@section
\scr@fnt@subsection
\scr@fnt@subsubsection
\scr@fnt@paragraph
\scr@fnt@subparagraph
minisec

Definition der Elemente der Gliederung, deren Schrift geändert werden kann:

```
⟨/class⟩
\newcommand*{\scr@fnt@disposition}{\sectfont}
\aliaskomafont{sectioning}{disposition}
⟨*class⟩
\newcommand*{\scr@fnt@part}{\size@part}
\newcommand*{\scr@fnt@partnumber}{\size@partnumber}
⟨book | report⟩\newcommand*{\scr@fnt@chapter}{\size@chapter}
⟨book | report⟩\newcommand*{\scr@fnt@chapterprefix}{\size@chapterprefix}
\newcommand*{\scr@fnt@section}{\size@section}
\newcommand*{\scr@fnt@subsection}{\size@subsection}
\newcommand*{\scr@fnt@subsubsection}{\size@subsubsection}
\newcommand*{\scr@fnt@paragraph}{\size@paragraph}
\newcommand*{\scr@fnt@subparagraph}{\size@subparagraph}
```

12.10 Definitionen für Präambeln

Bei den KOMA-Script-Klassen scrbook und scrreprt können `\part` und `\chapter` mit Präambeln versehen werden. Es wird gleichzeitig eine Präambel über und unter der jeweiligen Überschrift unterstützt.

\use@preamble

Dieses Makro dient allgemein zum Setzen einer Präambel (für Gliederungsüberschriften), wenn eine solche definiert ist. Als Argument wird der Name der Präambel übergeben, an den automatisch `@preamble` angehängt wird. Zum Schluss wird die gespeicherte Präambel gelöscht.

```
\newcommand*{\use@preamble}[1]{%
  \@ifundefined{#1@preamble}{}{%
    \@nameuse{#1@preamble}%
    \global\expandafter\let\csname#1@preamble\endcsname=\relax
  }%
}
```

\set@preamble

Das Makro `\set@preamble` dient allgemein der Defintion einer Präambel (für Gliederungsüberschriften). Als Argument wird der Name der Gliederungsebene übergeben. Danach folgt ein optionales Argument, das die Position der Präambel angibt. Darauf folgt ein weiteres optionales Argument, das die Breite der Präambel angibt. Ist nur ein optionales Argument angegeben, so ist es die Position. Es folgt ein obligatorisches Argument mit dem Inhalt der Präambel. Die Definition der Präambel erfolgt global.

\set@@preamble
\set@@@preamble
\set@@@@preamble

Da mehrere optionale Argumente nach einem obligatorischen Argument auszuwerten sind, werden diverse Hilfsmakros benötigt.

```
\newcommand*{\set@preamble}[1]{%
```

Zunächst wird das erste optionale Argument gesucht.

```
  \@ifnextchar [%]
  {\set@@preamble{#1}}{\set@@@preamble{#1}[][\hsize]}%
```

```
}
\newcommand*{\set@@preamble}{}
\def\set@@preamble#1[#2]{%
```

Wenn ein optionales Argument existiert, kann es auch ein zweites geben. Wenn nicht, wird mit der Voreinstellung gearbeitet.

```
  \@ifnextchar [%]
  {\set@@@preamble{#1}[{#2}]}{\set@@@preamble{#1}[{#2}][\hsize]}%
}
\newcommand{\set@@@preamble}{}
\long\def\set@@@preamble#1[#2][#3]#4{%
```

Dies ist das Hauptmakro, das die eigentliche Arbeit macht. Es muss `\long` definiert werden, da die Präambel auch aus mehreren Absätzen bestehen kann.

Zunächst wird das Positionsargument ausgewertet. Dies geschieht mit lokalen Hilfsmakros und einer Schleife. Zur Funktion der Schleife siehe im dokumentierten LaTeX-Kern.

```
  \begingroup
    \def\prmbl@pos{#2}\let\prmbl@hpos\relax\let\prmbl@vpos\relax
    \expandafter \@tfor \expandafter \@tempa
      \expandafter :\expandafter =\prmbl@pos
    \do{%
      \if \@tempa l%
        \set@preamble@hpos{0}{#2}%
      \fi%
      \if \@tempa c%
        \set@preamble@hpos{1}{#2}%
      \fi%
      \if \@tempa r%
        \set@preamble@hpos{2}{#2}%
      \fi%
      \if \@tempa u%
        \set@preamble@vpos{0}{#2}%
      \fi%
      \if \@tempa o%
        \set@preamble@vpos{1}{#2}%
      \fi%
    }%
```

Wurde keine horizontale oder vertikale Ausrichtung definiert, so wird nun der Standardwert dafür eingesetzt.

```
    \@ifundefined{prmbl@hpos}{\def\prmbl@hpos{0}}{}%
    \@ifundefined{prmbl@vpos}{\def\prmbl@vpos{0}}{}%
```

Nun sind die Positionen ermittelt und es werden abhängig davon entsprechende Päambelmakros definiert.

```
    \ifcase\prmbl@hpos
```

Zunächst links unten und oben:

```
      \ifcase\prmbl@vpos
        \set@@@@preamble{#1@u}{t}{#3}{}{\hfil}{#4}%
      \else
        \set@@@@preamble{#1@o}{b}{#3}{}{\hfil}{#4}%
      \fi
    \or
```

Dann zentriert unten und oben:

```
      \ifcase\prmbl@vpos
        \set@@@@preamble{#1@u}{t}{#3}{\hfil}{\hfil}{#4}%
      \else
        \set@@@@preamble{#1@o}{b}{#3}{\hfil}{\hfil}{#4}%
      \fi
    \else
```

Zum Schluss rechts und oben:

```
      \ifcase\prmbl@vpos
        \set@@@@preamble{#1@u}{t}{#3}{\hfil}{}{#4}%
      \else
        \set@@@@preamble{#1@o}{b}{#3}{\hfil}{}{#4}%
      \fi
    \fi
  \endgroup
}
```

Es folgt das Makro, mit dem die Definition tatsächlich stattfindet. Das erste Argument ist der Name der Präambel, das zweite die Positionsoption für die verwendete `\parbox`, das dritte ist die Breite der `\parbox`, dann zwei Argumente mit der horizontalen Ausrichtung vor und nach der `\parbox` und das sechste Agument ist schließlich der Inhalt der `\parbox`.

```
\newcommand{\set@@@@preamble}[6]{%
  \expandafter\gdef\csname #1@preamble\endcsname{%
    \hbox to\hsize{#4\parbox[{#2}]{#3}{#6\par}#5\par}%
  }%
}
```

`\set@preamble@hpos` Eines der Hilfsmakros dient dazu, die horizontale Position der Präambel einzustellen, falls dies nicht bereits erfolgt ist. Ist es bereits nicht übereinstimmend erfolgt, wird ein Fehler ausgegeben. Dieses Makro ist extrem intern und funktioniert nur im Kontext vor `\set@preamble`.

```
\newcommand*{\set@preamble@hpos}[2]{%
  \@ifundefined{prbml@hpos}{%
    \def\prmbl@hpos{#1}%
  }{%
    \ifnum \prmbl@hpos=0\relax\else%
      \ClassError{\KOMAClassName}{%
        inconsistent hpos options%
      }{%
        You said ‘#2’ as position option. But you have
        to\MessageBreak
```

```
        use only one from ‘l’, ‘r’ or ‘c’ not two of
        these.\MessageBreak
        If you’ll continue only first hpos option will be
        used%
      }%
    \fi
  }%
}
```

\set@preamble@vpos

Das Hilfsmakro für die vertikale Position funktioniert in gleicher Weise.

```
\newcommand*{\set@preamble@vpos}[2]{%
  \@ifundefined{prbml@vpos}{%
    \def\prmbl@vpos{#1}%
  }{%
    \ifnum \prmbl@vpos=0\relax\else%
      \ClassError{\KOMAClassName}{%
        inconsistent vpos options%
      }{%
        You said ‘#2’ as position option. But you have
        to\MessageBreak
        use only one from ‘o’ or ‘u’ not two of
        these.\MessageBreak
        If you’ll continue only first vpos option will be
        used%
      }%
    \fi
  }%
}
```

\setchapterpreamble
\chapter@u@preamble
\chapter@o@preamble

Diese Makros werden benötigt, um die Kapitel-Präambel zu setzen bzw. zu speichern.

```
\newcommand*{\setchapterpreamble}{\set@preamble{chapter}}
\newcommand*{\chapter@u@preamble}{}
\let\chapter@u@preamble\relax
\newcommand*{\chapter@o@preamble}{}
\let\chapter@o@preamble\relax
```

\setpartpreamble
\part@preamble@u
\part@preamble@o

Analog zu den Kapitel gibt es Befehle für eine Teil-Präambel.

```
\newcommand*{\setpartpreamble}{\set@preamble{part}}
\newcommand*{\part@preamble@u}{}
\let\part@preamble@u\relax
\newcommand*{\part@preamble@o}{}
\let\part@preamble@o\relax
```

12.11 Einträge ins Inhaltsverzeichnis

Die Gliederungsbefehle schreiben auch Einträge ins Inhaltsverzeichnis, die dort ebenfalls formatiert werden müssen.

\l@part Der Eintrag für \part:

```
\newcommand*\l@part[2]{%
  \ifnum \c@tocdepth >-2\relax
⟨article⟩     \addpenalty{\@secpenalty}%
⟨book | report⟩     \addpenalty{-\@highpenalty}%
    \addvspace{2.25em \@plus\p@}%
    \setlength{\@tempdima}{2em}%
    \if@tocleft
      \ifx\toc@l@number\@empty\else
        \setlength\@tempdima{0\toc@l@number}%
      \fi
    \fi
    \begingroup
      \parindent \z@ \rightskip \@pnumwidth
      \parfillskip -\@pnumwidth
      \leavevmode
      \advance\leftskip\@tempdima
      \hskip -\leftskip
      \usekomafont{partentry}{#1\nobreak
        \usekomafont{partentrypagenumber}{\hfil\nobreak
          \hb@xt@\@pnumwidth{\hss#2}}}\par
```

Aus unerfindlichen Gründen war es bis Version 2.96 erlaubt, dass zwischen der \part-Überschrift und der nachfolgenden Überschrift ein Umbruch stattfinden kann. Ab Version 2.96a gilt das nur noch, wenn die nachfolgende Überschrift wiederum eine \part-Überschrift ist.

```
      \ifnum \scr@compatibility>\@nameuse{scr@v@2.96}\relax
      \endgroup
      \penalty20010
      \else
⟨*article⟩
        \if@compatibility
          \global\@nobreaktrue
          \everypar{\global\@nobreakfalse\everypar{}}%
        \else
⟨/article⟩
          \penalty\@highpenalty
⟨article⟩          \fi
      \endgroup
    \fi
  \fi
}
```

\l@chapter Kapiteleinträge gibt es natürlich nicht bei scrartcl.

```
\newcommand*\l@chapter[2]{%
  \ifnum \c@tocdepth >\m@ne
```

```
    \ifnum \lastpenalty<20010
      \addpenalty{-\@highpenalty}%
    \fi
    \vskip 1.0em \@plus\p@
    \setlength\@tempdima{1.5em}%
    \if@tocleft
      \ifx\toc@l@number\@empty\else
        \setlength\@tempdima{0\toc@l@number}%
      \fi
    \fi
    \begingroup
      \raggedchapterentry
      \parindent \z@ \advance\rightskip \@pnumwidth
      \parfillskip -\@pnumwidth
      \interlinepenalty\@M
      \leavevmode
      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
      \usekomafont{chapterentry}{#1\nobreak
        \usekomafont{chapterentrypagenumber}{%
```

Und dann könnte man noch folgendes machen:

```
          \if@chapterentrywithdots
            \leaders\hbox{$\m@th
              \mkern \@dotsep mu\hbox{.}\mkern \@dotsep
              mu$}%
          \fi
```

```
          \hfill\nobreak
          \hb@xt@\@pnumwidth{\hss#2}}}\par
      \ifnum \scr@compatibility>\@nameuse{scr@v@2.96}\relax
      \endgroup
      \penalty20009
      \else
        \penalty\@highpenalty
      \endgroup
    \fi
  \fi
}
```

`\raggedchapterentry` Ausrichtung des Textes der Kapiteleinträge im Inhaltsverzeichnis.

```
\newcommand*{\raggedchapterentry}{}
```

`\l@chapteratlist` Dieses Makro wird ggf. für die Kapiteleinträge in den Listen der Gleitumgebungen verwendet. Zunächst einmal wird hier dasselbe verwendet, wie für die Kapiteleinträge ins Inhaltsverzeichnis.

```
\newcommand*{\l@chapteratlist}{\l@chapter}
```

`\l@section` Bei der Artikel-Klasse folgt die Formatierung für `\section`-Einträge, an Stelle der Einträge für `\chapter` der Buch- oder der Bericht-Klasse. Bei diesen handelt es sich hier stattdessen um eine normale Ebene.

```
⟨*article⟩
\newcommand*\l@section[2]{%
  \ifnum \c@tocdepth >\z@
    \ifnum \lastpenalty<20009
      \addpenalty{\@secpenalty}%
    \fi
    \addvspace{1.0em \@plus\p@}%
    \setlength\@tempdima{1.5em}%
     \if@tocleft
      \ifx\toc@l@number\@empty\else
        \setlength\@tempdima{0\toc@l@number}%
      \fi
    \fi
   \begingroup
      \raggedsectionentry
      \parindent \z@ \advance\rightskip \@pnumwidth
      \parfillskip -\@pnumwidth
      \interlinepenalty\@M
      \leavevmode
      \advance\leftskip \@tempdima \null\nobreak\hskip -\leftskip
      \usekomafont{sectionentry}{#1\nobreak
        \usekomafont{sectionentrypagenumber}{%
```

Und dann könnte man noch folgendes machen:

```
          \if@sectionentrywithdots
            \leaders\hbox{$\m@th
              \mkern \@dotsep mu\hbox{.}\mkern \@dotsep
              mu$}%
          \fi
```

```
          \hfill\nobreak
          \hb@xt@\@pnumwidth{\hss#2}}}\par
    \endgroup
    \ifnum \scr@compatibility>\@nameuse{scr@v@2.96}\relax
      \penalty20008
    \fi
  \fi
}
```

`\raggedsectionentry` Ausrichtung des Textes der Abschnittseinträge im Inhaltsverzeichnis.

```
\newcommand*{\raggedsectionentry}{}
```

```
⟨/article⟩
⟨*report | book⟩
\newcommand*\l@section{\bprot@dottedtocline{1}{1.5em}{2.3em}}
⟨/report | book⟩
```

\l@subsection
\l@subsubsection
\l@paragraph
\l@subparagraph

Die übrigen Ebenen werden normal behandelt.

```
⟨*article⟩
\newcommand*\l@subsection{\bprot@dottedtocline{2}{1.5em}{2.3em}}
\newcommand*\l@subsubsection{\bprot@dottedtocline{3}{3.8em}{3.2em}}
\newcommand*\l@paragraph{\bprot@dottedtocline{4}{7.0em}{4.1em}}
\newcommand*\l@subparagraph{\bprot@dottedtocline{5}{10em}{5em}}
⟨/article⟩
⟨*report | book⟩
\newcommand*\l@subsection{\bprot@dottedtocline{2}{3.8em}{3.2em}}
\newcommand*\l@subsubsection{\bprot@dottedtocline{3}{7.0em}{4.1em}}
\newcommand*\l@paragraph{\bprot@dottedtocline{4}{10em}{5em}}
\newcommand*\l@subparagraph{\bprot@dottedtocline{5}{12em}{6em}}
⟨/report | book⟩
```

\bprot@dottedtocline

Dieses Makro wird benötigt, um die Ebenen unterhalb von `\chapter` bzw. `\section` im Inhaltsverzeichnis ebenfalls vor unerwünschten Umbrüchen zwischen Eintrag und Unterebene zu schützen.

```
\newcommand*{\bprot@dottedtocline}[5]{%
  \ifnum \scr@compatibility>\@nameuse{scr@v@2.96}\relax
    \ifnum #1>\c@tocdepth \else
      \begingroup
        \@tempcnta 20010
        \advance \@tempcnta by -#1
      \ifnum \@tempcnta>\lastpenalty
        \endgroup
        \addpenalty{\@lowpenalty}%
      \else
        \endgroup
      \fi
      \@dottedtocline{#1}{#2}{#3}{#4}{#5}%
      \@tempcnta 20009
      \advance \@tempcnta by -#1
      \penalty\@tempcnta
    \fi
  \else
    \@dottedtocline{#1}{#2}{#3}{#4}{#5}%
  \fi
}
```

Datei m
scrkliof.dtx

13 Verzeichnisse

LaTeX Klassen bieten eine ganze Reihe von Verzeichnissen. Thema dieses Abschnitts sind die Verzeichnisse der Gleitumgebungen und das Inhaltsverzeichnis. KOMA-Script bietet selbst das Verzeichnis der Gleittabellen und der Abbildungen. Es unterstützt außerdem die Verzeichnisse der mit dem float Paket angelegten Gleitumgebungen.

Briefe verfügen über keine Verzeichnisse.

13.1 Optionen für das Inhaltsverzeichnis

Einige Anweisungen und Einstellungen werden bereits für die Optionen benötigt und deshalb vorab definiert.

`\if@tocleft` `\@toclefttrue` `\@tocleftfalse`

Die Wahl der Darstellungsart für das Inhaltsverzeichnis wird in einem Schalter gespeichert.

```
\newif\if@tocleft
```

`toc`

Dies ist die zentrale Option für Einstellungen des Inhaltsverzeichnisses.

```
\KOMA@key{toc}{%
  \KOMA@set@ncmdkey{toc}{@tempa}{%
```

Es gibt Werte, um zu bestimmen, welche Verzeichnisse ins Inhaltsverzeichis aufgenommen werden. Eigentlich sind das Optionen der jeweiligen Verzeichnisses, aber das muss sich der Anwender ja nicht unbedingt merken.

```
    {noindex}{0},{noidx}{0},%
    {index}{1},{idx}{1},%
    {nolistof}{2},%
    {listof}{3},%
    {listofnumbered}{4},{numberedlistof}{4},%
    {nobibliography}{5},{nobib}{5},%
    {bibliography}{6},{bib}{6},%
    {bibliographynumbered}{7},{bibnumbered}{7},{numberedbibliography}{7},%
    {numberedbib}{7},%
```

Dann gibt es Optionen für die Darstellungsarten des Inhaltsverzeichnisses. Die Variante `graduated` ist die altbekannte Variante. Bei `flat` hingegen wird eine tabellenartige Form verwendet, bei der die Nummern, Texte und Seitenzahlen jeweils untereinander stehen. Der für die Nummern benötigte Platz wird dabei automatisch ermittelt.

```
    {flat}{8},{left}{8},%
    {graduated}{9},{indent}{9},{indented}{9}%
```

```
  }{#1}%
  \ifcase \@tempa\relax
    \KOMA@options{index=notoc}{}%
  \or
    \KOMA@options{index=totoc}{}%
  \or
    \KOMA@options{listof=notoc}{}%
  \or
    \KOMA@options{listof=totoc}{}%
  \or
    \KOMA@options{listof=numbered}{}%
  \or
    \KOMA@options{bibliography=nottotoc}{}%
  \or
    \KOMA@options{bibliography=totoc}{}%
  \or
    \KOMA@options{bibliography=totocnumbered}{}%
  \or
    \@toclefttrue
  \or
    \@tocleftfalse
  \fi
}
```

tocleft
tocindent

```
\KOMA@DeclareDeprecatedOption{tocleft}{toc=flat}
\KOMA@DeclareDeprecatedOption{tocindent}{toc=graduated}
```

13.2 Optionen für die Gleitumgebungen

Einige Anweisungen und Einstellungen werden bereits für die Optionen benötigt und deshalb vorab definiert.

\float@@listhead

Setzt nur die Überschrift und sonst nichts. Dabei wird die Überschrift als einziges Argument übergeben.

```
\newcommand*{\float@@listhead}{%
⟨article⟩   \section*
⟨report | book⟩   \chapter*
}
```

\if@dynlist
\@dynlisttrue
\@dynlistfalse

Die Wahl der Darstellungsart wird in einem Schalter gespeichert.

```
\newif\if@dynlist
```

listof

Dies ist die zentrale Option für die Gleitumgebungsverzeichnisse.

```
\KOMA@key{listof}{%
  \KOMA@set@ncmdkey{listof}{@tempa}{%
```

Es gibt Werte, die dafür sorgen, dass die Verzeichnisse der Gleitumgebungen im Inhalsverzeichnis aufgeführt werden. Dazu muss nur `\float@@listhead` umdefiniert werden.

```
    {notoc}{0},{nottotoc}{0},{plainheading}{0},%
    {totoc}{1},{toc}{1},{notnumbered}{1},%
    {numbered}{2},{totocnumbered}{2},{tocnumbered}{2},{numberedtotoc}{2},%
    {numberedtoc}{2},%
```

Dann gibt es Optionen für die Darstellungsarten des Inhaltsverzeichnisses. Die Variante `graduated` ist die altbekannte Variante. Bei `flat` hingegen wird eine tabellenartige Form verwendet, bei der die Nummern, Texte und Seitenzahlen jeweils untereinander stehen. Der für die Nummern benötigte Platz wird dabei automatisch ermittelt.

```
    {flat}{3},{left}{3},%
    {graduated}{4},{indent}{4},{indented}{4},%
```

Dann gibt es eine Option, mit der die Gliederungsebene der Verzeichnisse verändert werden kann.

```
    {leveldown}{5},
```

Dann gibt es eine Option, mit der die Einträge in die Verzeichnisse mit einem Prefix versehen werden können, soweit ein solcher Prefix existiert. Dieser Wert beinhaltet gleichzeitig auch den Wert `flat`.

```
    {entryprefix}{6},
```

Dann gibt es Optionen, für die Unterteilung auf Kapitelebene. Allerdings nur für scrbook und scrreprt.

```
⟨*book | report⟩
    {chapterentry}{7},{withchapterentry}{7},%
    {nochaptergap}{8},{ignorechapter}{8},%
    {chaptergapsmall}{9},{smallchaptergap}{9},%
    {chaptergapline}{10},{onelinechaptergap}{10}%
⟨/book | report⟩
  }{#1}%
  \ifcase \@tempa\relax
    \renewcommand*{\float@@listhead}{%
⟨article⟩      \section*
⟨report | book⟩      \chapter*
    }%
    \doforeachtocfile[float]{%
      \unsettoc{\@currext}{numbered}%
      \unsettoc{\@currext}{totoc}%
    }%
    \AtAddToTocList[float]{%
      \unsettoc{\@currext}{numbered}%
      \unsettoc{\@currext}{totoc}%
    }%
  \or% totoc
    \renewcommand*{\float@@listhead}{%
```

```
⟨article⟩      \addsec
⟨report | book⟩      \addchap
    }%
    \doforeachtocfile[float]{%
      \unsettoc{\@currext}{numbered}%
      \setuptoc{\@currext}{totoc}%
    }%
    \AtAddToTocList[float]{%
      \unsettoc{\@currext}{numbered}%
      \setuptoc{\@currext}{totoc}%
    }%
  \or% numbered
    \renewcommand*{\float@@listhead}{%
⟨article⟩      \section
⟨report | book⟩      \chapter
    }%
    \doforeachtocfile[float]{%
      \setuptoc{\@currext}{numbered}%
      \setuptoc{\@currext}{totoc}%
    }%
    \AtAddToTocList[float]{%
      \setuptoc{\@currext}{numbered}%
      \setuptoc{\@currext}{totoc}%
    }%
  \or% flat
    \@dynlisttrue
  \or% graduated
    \@dynlistfalse
  \or% leveldown
    \doforeachtocfile[float]{%
      \setuptoc{\@currext}{leveldown}%
    }%
    \AtAddToTocList[float]{%
      \setuptoc{\@currext}{leveldown}%
    }%
  \or% entryprefix
    \@dynlisttrue
    \doforeachtocfile[float]{%
      \BeforeStartingTOC[\@currext]{%
        \scr@ifundefinedorrelax{listof\@currext entryname}{}{%
          \expandafter\def\expandafter\numberline@prefix
          \expandafter{\csname listof\@currext entryname\endcsname
            \nobreakspace}%
        }%
      }%
    }%
    \AtAddToTocList[float]{%
      \BeforeStartingTOC[\@currext]{%
        \scr@ifundefinedorrelax{listof\@currext entryname}{}{%
```

```
          \expandafter\def\expandafter\numberline@prefix
          \expandafter{\csname listof\@currext entryname\endcsname
            \nobreakspace}%
        }%
      }%
    }%
⟨*book | report⟩
  \or% chapterentry
    \KOMA@options{chapteratlists=entry}{}%
  \or% chaptergap
    \KOMA@options{chapteratlists=\z@}{}%
  \or% chaptergapsmall
    \KOMA@options{chapteratlists=10\p@}{}%
  \or% chaptergapline
    \KOMA@options{chapteratlists=\baselineskip}{}%
⟨/book | report⟩
  \fi
}
```

liststotoc

```
\KOMA@DeclareDeprecatedOption{liststotoc}{listof=totoc}
```

liststotocnumbered

```
\KOMA@DeclareDeprecatedOption{liststotocnumbered}{listof=numbered}
```

listsleft
listsindent

```
\KOMA@DeclareDeprecatedOption{listsleft}{listof=flat}
\KOMA@DeclareDeprecatedOption{listsindent}{listof=graduated}
```

chapteratlists
\if@chaptertolists
\@chapterlistsgap

Option, um einzugstellen, ob Kapitel in den Verzeichnissen der Gleitumgebungen Spuren hinterlassen. Man kann wahlweise Lücken einstellen (Voreinstellung mit 10 pt) oder die Kapitelüberschrift eintragen lassen. Der Schalter gibt dabei an, ob die Kapitelüberschrift eingetragen werden soll. Das interne Hilfsmakro `\@chapterlistsgap` enthält den gewünschten Abstand.

```
⟨*book | report⟩
\newcommand*{\@chapterlistsgap}{10\p@}
\newif\if@chaptertolists\@chaptertolistsfalse
\KOMA@key{chapteratlists}[entry]{%
  \ifstr{#1}{entry}{%
    \@chaptertoliststrue\renewcommand*{\@chapterlistsgap}{\z@}%
  }{%
    \@chaptertolistsfalse\renewcommand*{\@chapterlistsgap}{#1}%
  }%
}
⟨/book | report⟩
```

13.3 Definitionen für alle Verzeichnisse

Einige Definitionen werden von unterschiedlichen Verzeichnissen verwendet. Diese sind hier zu finden.

Bei Büchern wird für jedes Verzeichnis des Besitzers `float` das Feature `chapteratlist` gesetzt. Das gilt auch für solche Verzeichnisse, die noch gar nicht definiert sind.

```
⟨*book | report⟩
\doforeachtocfile[float]{%
  \setuptoc{\@currext}{chapteratlist}%
}
\AtAddToTocList[float]{%
  \setuptoc{\@currext}{chapteratlist}%
}
⟨/book | report⟩
```

`\numberline` Dieses Makro wird indirekt im Inhaltsverzeichnis verwendet und muss umdefiniert werden, um Duden, 20. Auflage, Regel 5 und Regel 6 zu implementieren.

```
\renewcommand*{\numberline}[1]{%
  \numberline@box{%
    \numberline@prefix\numberline@numberformat{#1}\numberline@postfix
  }%
}
```

`\numberline@prefix` Das ist in der Voreinstellung einfach leer. Es könnte verwendet werden, um etwas vor die Nummer zu setzen oder die Nummer rechtsbündig zu setzen.

```
\newcommand*{\numberline@prefix}{}
```

`\numberline@postfix` Das ist in der Voreinstellung einfach `\hfil` und dient dazu die Nummer linksbündig zu platzieren.

```
\newcommand*{\numberline@postfix}{\hfil}
```

`\numberline@numberformat` Das dient in der Voreinstellung dazu, Regel 5 und 6 aus der 20. Auflage des DUDENs umzusetzen.

```
\newcommand*{\numberline@numberformat}[1]{#1\autodot}
```

`\numberline@box` Das dient in der Voreinstellung dazu, die Gliederungsnummer in eine Box zu setzen.

```
\newcommand*{\numberline@box}[1]{\hb@xt@\@tempdima{#1}}
```

`\@pnumwidth` `\@tocrmarg` `\@dotsep` Einige Makros für die Formatierung der Verzeichniseinträge:

`\@pnumwidth` ist der Platz für die Seitenzahl.

`\@tocrmarg` ist der rechte Rand bei mehrzeiligen Verzeichniseinträgen. Dieser wird hier größer als `\@pnumwidth` gewählt.

\dotsep ist der Abstand zwischen den einzelnen Punkten.

```
\newcommand*\@pnumwidth{1.55em}
\newcommand*\@tocrmarg {2.55em}
\newcommand*\@dotsep{4.5}
```

\before@starttoc Dieses Makro führt im Falle von tocleft oder listsleft am Anfang eines Verzeichnisses alle notwendigen Änderungen durch, um das Verzeichnis entsprechend zu handhaben.

\scr@dottedtocline
\scr@numberline
\last@l@number

Dafür werden ein paar Hilfsmakros benötigt, die vorsichtshalber global vordefiniert werden.

```
\newcommand*{\scr@dottedtocline}{}
\newcommand*{\last@l@number}{}
```

\@l@number In diesem Makro wird während der Erstellung eines Verzeichnisses Inhaltsverzeichnisses die aktuelle Breite der Nummer gespeichert.

\set@l@number Dieses Makro ist dafür verantwortlich, dass zum einen \@tempdima für den Aufruf von \numberline@box auf den richtigen Wert gesetzt wird, zum anderen wird die aktuelle größte Breite der Nummer hier angepasst.

```
\newcommand*{\@l@number}{}
\newcommand*{\set@l@number}[1]{%
  \settowidth{\@tempdima}{#1\enskip}%
  \ifdim\@tempdima >\@l@number
    \xdef\@l@number{\the\@tempdima}%
  \fi%
  \ifdim\@tempdima >\last@l@number \else
    \setlength{\@tempdima}{\last@l@number}%
  \fi%
}
```

Kommen wir endlich zu \before@starttoc:

```
\newcommand*{\before@starttoc}[1]{
  \let\scr@dottedtocline=\@dottedtocline
  \renewcommand*{\@dottedtocline}[3]{%
    \scr@dottedtocline{##1}{\z@}{\last@l@number}}%
  \renewcommand*{\numberline@box}[1]{%
    \set@l@number{##1}\hb@xt@\@tempdima{##1}}%
  \gdef\@l@number{\z@}%
  \@ifundefined{#1@l@number}{%
    \def\last@l@number{2em}%
  }{%
    \expandafter\ifx\csname #1@l@number\endcsname\@empty
      \let\last@l@number\z@
    \else
      \expandafter\let\expandafter\last@l@number
      \expandafter=\csname#1@l@number\endcsname
    \fi
```

```
  }%
}
```

\after@starttoc Dieses Makro schreibt am Ende den entsprechenden Eintrag in die aux-Datei.

```
\newcommand*{\after@starttoc}[1]{%
  \protected@write\@auxout{}{%
    \string\gdef\expandafter\string\csname#1@l@number\endcsname{%
      \@l@number}}%
  }
```

Dafür sorgen, dass die Verzeichnisse von tocbasic so gesetzt werden, wie wir das gerne hätten (wird über einen Trick auch für das Inhaltsverzeichnis verwendet):

```
\g@addto@macro\tocbasic@@before@hook{%
  \if@dynlist\expandafter\before@starttoc\expandafter{\@currext}\fi
}
\g@addto@macro\tocbasic@@after@hook{%
  \if@dynlist\expandafter\after@starttoc\expandafter{\@currext}\fi
}
```

13.4 Definitionen für das Inhaltsverzeichnis

\contentsname Der Name des Verzeichnisses.

\listoftocname

```
\newcommand*\contentsname{Contents}
\newcommand*\listoftocname{\contentsname}
```

tocdepth Der Zähler gibt an, bis zu welcher Gliederungebene Einträge in das Inhaltsverzeichnis aufgenommen werden. Da scrartcl eine Ebene tiefer beginnt, wird auch eine Ebene tiefer eingetragen. Damit finden sich – abgesehen von \part – immer drei Ebenen im Inhaltsverzeichnis.

```
⟨book | report⟩\setcounter{tocdepth}{2}
⟨article⟩\setcounter{tocdepth}{3}
```

\toc@heading Befehl, zum Setzen der Überschrift des Inhaltsverzeichnisses. Eigentlich ist das überflüssig, weil es nur einmal verwendet wird. Aus Gründen der Konsistenz mit den anderen Verzeichnissen machen wir das hier aber genauso.

```
\newcommand*\toc@heading{%
  \ClassWarning{\KOMAClassName}{%
    usage of deprecated \string\toc@heading!\MessageBreak
    You should use the features of package ‘tocbasic’\MessageBreak
    instead of \string\toc@heading.\MessageBreak
    Definition of \string\toc@heading\space my be removed from\MessageBreak
    KOMA-Script soon, so it should not be used
  }%
⟨article⟩  \section*{\contentsname}%
⟨book | report⟩  \chapter*{\contentsname}%
```

```
  \@mkboth{\contentsname}{\contentsname}%
}
```

\toc@l@number Dieses Makro wird innerhalb von \tableofcontents über \before@starttoc und \after@starttoc in der aux-Datei global definiert. Aus Sicherheitsgründen wird es hier global vordefiniert.

```
\newcommand*{\toc@l@number}{}
```

\tableofcontents Die Ausgabe des Inhaltsverzeichnisses.

```
\newcommand*{\tableofcontents}{%
  \begingroup\let\if@dynlist\if@tocleft\listoftoc{toc}\endgroup
}
```

\addtocentrydefault Seit Version 3.08 werden Einträge ins Inhaltsverzeichnis in der Voreinstellung über diese Anweisung vorgenommen. Das erste Argument ist dabei die Gliederungsebene. Das zweite Argument ist die (formatierte) Gliederungsnummer oder leer, falls die Überschrift nicht nummeriert wird. Das dritte Argument ist der Überschriftentext, der in das Inhaltsverzeichnis soll.

```
\newcommand{\addtocentrydefault}[3]{%
  \ifstr{#2}{}{%
    \addcontentsline{toc}{#1}{#3}%
  }{%
    \addcontentsline{toc}{#1}{\protect\numberline{#2}#3}%
  }%
}
```

13.5 Definitionen für die Verzeichnisse der Gleitumgebungen

\float@listhead In Absprache mit Anselm Lingnau, dem Autor des Pakets float, wird ab Version 2.8b \float@headings zum Setzen des Kolumnentitels bei den Verzeichnissen verwendet, die float für neu definierte floats zur Verfügung stellt. Dabei definiert float die Anweisung nur, wenn sie nicht bereits existiert. Die Zusammenarbeit wird dadurch verbessert. Da Anselm sich in letzter Minute entschlossen hat, das Makro \float@listhead zu nennen, wurde dies in Version 2.8g angepasst.

```
\newcommand*{\float@listhead}[1]{%
  \scr@float@listhead@warning
  \float@@listhead{#1}%
  \@mkboth{#1}{#1}%
```

Dies ist der passende Ort, um die Breite, die für die Nummer des Gleitobjekts benötigt wird zu ermitteln und anzupassen, falls dies geünscht wird. Dazu findet eine lokale Umdefinierung von \@starttoc statt. Die Umdefinierung ist deshalb lokal, weil alle mir bekannten Verzeichnisse von Gleitobjekten innerhalb einer Gruppe ausgegeben werden.

```
  \if@dynlist%
    \newcommand*{\scr@starttoc}{}%
```

```
    \let\scr@starttoc=\@starttoc
    \renewcommand*{\@starttoc}[1]{%
      \before@starttoc{##1}\scr@starttoc{##1}\after@starttoc{##1}%
    }%
  \fi
}
```

\scr@float@listhead@warning Die Warnung für die Verwendung von \float@listhead, dessen Definition irgendwann aus den Klassen verschwinden wird.

```
\newcommand*{\scr@float@listhead@warning}{%
  \ClassWarning{\KOMAClassName}{%
    Usage of deprecated \string\float@listhead!\MessageBreak
    You should use the features of package ‘tocbasic’\MessageBreak
    instead of \string\float@listhead.\MessageBreak
    Definition of \string\float@listhead\space my be removed from\MessageBreak
    ‘\KOMAClassName’ soon, so it should not be used
  }%
}
```

\scr@float@addtolists@warning Die Warning für die Verwendung von \float@addtolists, dessen Unterstützung irgendwann aus den Klassen verschwinden wird.

```
\newcommand*{\scr@float@addtolists@warning}{%
  \ClassWarningNoLine{\KOMAClassName}{%
    \string\float@addtolists\space detected!\MessageBreak
    You should use the features of package ‘tocbasic’\MessageBreak
    instead of \string\float@addtolists.\MessageBreak
    Support for \string\float@addtolists\space may be removed
    from\MessageBreak
    ‘\KOMAClassName’ soon
  }%
  \global\let\scr@float@addtolists@warning\relax
}
```

\listfigurename Die Namen der Verzeichnisse auch für das Paket tocbasic.
\listtablename
\listoflofname
\listotlofname

```
\newcommand*\listfigurename{List of Figures}
\newcommand*\listoflofname{\listfigurename}
\newcommand*\listoflofentryname{\figurename}
\newcommand*\listtablename{List of Tables}
\newcommand*\listoflotname{\listtablename}
\newcommand*\listoflotentryname{\tablename}
```

\listoffigures Die Ausgabe des Abbildungsverzeichnisses.

```
\newcommand*\listoffigures{\listoftoc{lof}}
```

\l@figure Befehl zur Formatierung eines Verzeichniseintrags.

```
\newcommand*\l@figure{\@dottedtocline{1}{1.5em}{2.3em}}
```

\listoftables Die Ausgabe des Tabellenverzeichnisses.

```
\newcommand*\listoftables{\listoftoc{lot}}
```

`\l@table` Befehl zur Formatierung eines Verzeichniseintrags.

```
288 \let\l@table\l@figure
```

Datei n
scrkbib.dtx

14 Literaturverzeichnis

Das Literaturverzeichnis unterscheidet sich bei KOMA-Script nicht wesentlich von den Standardklassen.

Briefe haben kein Literaturverzeichnis.

14.1 Anweisungen, die für die Optionen benötigt werden

Einige Anweisungen werden über Optionen verändert. Dazu müssen sie aber bereits vor der Abarbeitung der Optionen definiert sein.

14.1.1 Literaturverzeichnisstile

`\newbibstyle` Neue Literaturverzeichnisstile müssen mit dieser Anweisung definiert werden, wenn sie von der Fehlermeldung korrekt angezeigt werden sollen. Das erste, optionale Argument ist der Name eines Elternstils. Das zweite Argument ist der Name des Stils. Das dritte Argument ist Code für den Stil. Darin sollten normalerweise Umdefinierungen für `\@openbib@code`, `\newblock`, `\bib@beginhook`, `\bib@endhook` enthalten sein. Die genannten Makros werden am Anfang des Stils zurückgesetzt. Danach wird der Elternstil geladen und dann die eigenen Definitionen.

```
\newcommand{\newbibstyle}[3][]{%
  \scr@ifundefinedorrelax{scr@bibstyle@#2}{%
    \scr@ifundefinedorrelax{scr@bibstyle@#1}{%
      \ClassError{\KOMAClassName}{%
        parent bibliography style ‘#1’ does not exist}{%
        You’ve told me to define the new bibliography style ‘#2’\MessageBreak
        to be a child of bibliography style ‘#1’, but bibliography
        style\MessageBreak
        ‘#1’ does not exist.\MessageBreak
        You have to remove the parent declaration or use an existing parent!
        See the KOMA-Script manual for more information.%
      }%
    }{%
      \@namedef{scr@bibstyle@#2}{%
        \let\@openbib@code\@empty
        \let\newblock\@empty
        \let\bib@beginhook\@empty
        \let\bib@endhook\@empty
        \csname scr@bibstyle@#1\endcsname
        #3%
      }%
```

```
      \g@addto@macro\scr@bibstylelist{, `#2'}%
    }%
  }{%
    \ClassError{\KOMAClassName}{bibliography style `#2' already exists}{%
      You've told me to define the new bibliography style `#2',\MessageBreak
      but this style already exists.\MessageBreak
      You have to use another bibliography style name.\MessageBreak
      See the KOMA-Script manual for more information.%
    }%
  }%
}
```

`\scr@bibstyle@` Nur ein Dummy, wenn kein Elternstil existiert.

```
\newcommand*{\scr@bibstyle@}{}
```

`\bib@beginhook` `\bib@endhook` Die beiden Haken, wie sie durch den Literaturverzeichnisstil vorgegeben werden.

```
\newcommand*{\bib@beginhook}{}
\newcommand*{\bib@endhook}{}
```

`\scr@bibstylelist` Liste der mit `\newbibstyle` definierten und daher bekannten Literaturverzeichnisstile.

`\@openbib@code` Code, der am Anfang des Literaturverzeichnisses ausgeführt wird und normalerweise ursprünglich für die `openbib`-Funktionalität verwendet wurde. Inzwischen wird er für die komplette Stilauswahl verwendet.

`\newblock`

```
\newcommand*{\scr@bibstylelist}{\@gobble}
\newcommand*{\@openbib@code}{}
\newcommand*{\newblock}{}
```

`\scr@bibstyle@openstyle` Vordefiniert sind zwei Stile. Mit dem Bibliographystil `openstyle` kann eine alternative Darstellung der Literatureinträge aktiviert werden. Dabei fügt `\newblock` keinen horizontalen Abstand mehr ein, sondern beginnt einen neuen Absatz (in Form einer neuen Zeile).

`\scr@bibstyle@oldstyle` Der Stil `oldstyle` entspricht hingegen der Voreinstellung.

```
\newbibstyle{openstyle}{%
  \renewcommand*{\@openbib@code}{%
    \advance\leftmargin\bibindent
    \itemindent -\bibindent
    \listparindent \itemindent
    \parsep \z@
  }%
  \renewcommand*{\newblock}{\par}%
}
\newbibstyle{oldstyle}{%
  \renewcommand*{\newblock}{\hskip .11em\@plus.33em\@minus.07em}%
```

```
}
\scr@bibstyle@oldstyle
```

14.1.2 Überschrift des Literaturverzeichnisses

`\bib@heading` Befehl zum Setzen der Überschrift des Literaturverzeichnisses.

```
\newcommand*\bib@heading{%
⟨*article⟩
  \section*{\refname}%
  \@mkboth{\refname}{\refname}%
⟨/article⟩
⟨*book | report⟩
  \chapter*{\bibname}%
  \@mkboth{\bibname}{\bibname}%
⟨/book | report⟩
}
```

14.2 Optionen für das Literaturverzeichnis

`bibliography` Dies ist der Generalschlüssel für alle Optionen, die das Literaturverzeichnis betreffen. Neu ist, dass verschiedene Stile definiert werden können. Desweiteren kann man `totoc` und `totocnumbered` wieder mit `nottotoc` aufgeben.

```
\KOMA@key{bibliography}{%
  \scr@ifundefinedorrelax{scr@bibstyle@#1}{%
    \let\FamilyElseValues\scr@bibstylelist
    \KOMA@set@ncmdkey{bibliography}{@tempa}{%
      {notoc}{0},{nottotoc}{0},{plainheading}{0},%
      {totoc}{1},{toc}{1},{notnumbered}{1},%
      {numbered}{2},{totocnumbered}{2},{tocnumbered}{2},{numberedtotoc}{2},%
      {numberedtoc}{2}%
    }{#1}%
    \ifcase \@tempa\relax
      \renewcommand*\bib@heading{%
⟨*article⟩
        \section*{\refname}%
        \@mkboth{\refname}{\refname}%
⟨/article⟩
⟨*book | report⟩
        \chapter*{\bibname}%
        \@mkboth{\bibname}{\bibname}%
⟨/book | report⟩
      }%
    \or% totoc
      \renewcommand*\bib@heading{%
⟨*article⟩
        \addsec{\refname}%
        \@mkboth{\refname}{\refname}%
⟨/article⟩
```

```
⟨*book | report⟩
        \addchap{\bibname}%
        \@mkboth{\bibname}{\bibname}%
⟨/book | report⟩
      }%
    \or% numbered
      \renewcommand*\bib@heading{%
⟨*article⟩
        \section{\refname}%
        \@mkboth{\sectionmarkformat \refname}{\sectionmarkformat \refname}%
⟨/article⟩
⟨*book | report⟩
        \chapter{\bibname}%
        \@mkboth{\chaptermarkformat \bibname}{\chaptermarkformat \bibname}%
⟨/book | report⟩
      }
    \fi
  }{%
    \csname scr@bibstyle@#1\endcsname
  }%
}
```

`openbib` Mit dieser Option kann eine alternative Darstellung der Literatureinträge aktiviert werden. Dabei fügt `\newblock` keinen horizontalen Abstand mehr ein, sondern beginnt einen neuen Absatz (in Form einer neuen Zeile).

```
\KOMA@DeclareStandardOption{openbib}{bibliography=openstyle}
```

`bibtotoc` `bibtotocnumbered` Die Option `bibtotoc` sorgt dafür, dass das Literaturverzeichnis im Inhaltsverzeichnis aufgeführt wird. Demgegenüber erweitert die Option `bibtotocnumbered` zu einem normalen Kapitel einschließlich Nummerierung.

```
\KOMA@DeclareDeprecatedOption{bibtotoc}{bibliography=totoc}
\KOMA@DeclareDeprecatedOption{bibtotocnumbered}{bibliography=totocnumbered}
```

14.3 Standardbezeichner

`\refname` `\bibname` Je nach Klasse wird `\refname` oder `\bibname` verwendet.

```
⟨article⟩\newcommand*\refname{References}
⟨book | report⟩\newcommand*\bibname{Bibliography}
```

14.4 Definitionen für das Literaturverzeichnis

`\setbibpreamble` `\bibpreamble` `\bibpreamble` kann vom Benutzer mit `\setbibpreamble` auf einen Text gesetzt werden, der nach der Überschrift eingefügt wird. Dasselbe Makro wird von natbib als `\relax` definiert; aus Kompatibilitätsgründen wird hier vergleichbar verfahren und `\let` bzw. `\def` statt `\renewcommand` verwendet. Außerdem wird ab Version 2.7j die Präambel nicht mehr in `\bib@heading` gesetzt, sondern in der `thebibliography`-Umgebung. Dies dient ebenfalls

der Kompatibilität mit natbib. Trotz aller Kompatibilität ist `\bibpreamble` als internes Makro zu betrachten.

```
\newcommand*{\bibpreamble}{\relax}
\let\bibpreamble\relax
\newcommand{\setbibpreamble}[1]{\def\bibpreamble{#1}}
```

`bibindent` Der Einzug der Einträge bei `openbib`. Voreingestellt sind 1,5 em.

```
\newdimen\bibindent
\bibindent=1.5em
```

`thebibliography` Nun werden noch die Umgebung für das Literaturverzeichnis und der Befehl für die einzelnen Einträge definiert.

```
\newenvironment{thebibliography}[1]{%
  \if@bib@break@
    \bib@beginhook
  \else
⟨*book | report⟩
    \ifx\bibpreamble\relax\else\ifx\bibpreamble\@empty\else
        \setchapterpreamble{\bibpreamble}%
    \fi\fi
⟨/book | report⟩
    \bib@heading
⟨*article⟩
    \ifx\bibpreamble\relax\else\ifx\bibpreamble\@empty\else
        \noindent\bibpreamble\par\nobreak
    \fi\fi
⟨/article⟩
    \bib@beginhook
    \bib@before@hook
  \fi
```

`\BreakBibliography` An dieser Stelle wird `\BreakBibliography` passend umdefiniert.

```
  \renewcommand{\BreakBibliography}[1]{%
    \@bib@break
    \end{thebibliography}%
    \begingroup ##1\endgroup
    \begin{thebibliography}{#1}
  }%

% Hier geht es jetzt mit \texttt{thebibliography} weiter.
  \list{%
    \@biblabel{\@arabic\c@enumiv}%
  }{%
    \settowidth\labelwidth{\@biblabel{#1}}%
    \leftmargin\labelwidth
    \advance\leftmargin\labelsep
    \@openbib@code
    \if@bib@break@
      \edef\@tempa{%
```

```
        \noexpand\usecounter{enumiv}%
        \noexpand\setcounter{enumiv}{\the\c@enumiv}%
      }\@tempa
      \no@bib@break
    \else
      \usecounter{enumiv}%
    \fi
    \let\p@enumiv\@empty
    \renewcommand*\theenumiv{\@arabic\c@enumiv}%
  }%
  \sloppy\clubpenalty4000 \widowpenalty4000
  \sfcode`\.=\@m
}{%
  \if@bib@break@
    \def\@noitemerr{%
      \@latex@warning{%
        \string\BreakBibliography\space at begin of `thebibliography'
        environment\MessageBreak
        or following \string\BreakBibliography}%
    }%
  \else
    \def\@noitemerr{%
      \@latex@warning{%
        Empty `thebibliography' environment or
        \string\BreakBibliography\MessageBreak
        before end of `thebibliography' environment}%
    }%
  \fi
  \endlist
  \if@bib@break@
  \else
    \bib@after@hook
    \let\bib@before@hook\@empty
    \let\bib@after@hook\@empty
  \fi
  \bib@endhook
}
```

`\if@bib@break@` `\no@bib@break` `\@bib@break`

Dieser Hilfsschalter arbeitet immer global!

```
\newif\if@bib@break@
\newcommand{\no@bib@break}{\global\let\if@bib@break@\iffalse}
\newcommand*{\@bib@break}{\global\let\if@bib@break@\iftrue}
```

`\bib@before@hook` `\bib@after@hook`

```
\newcommand*{\bib@before@hook}{}
\newcommand*{\bib@after@hook}{}
```

`\BreakBibliography`

Diese Anweisung unterbricht normalerweise das Literaturverzeichnis für ein paar eigene Anweisungen (z. B. Unterüberschriften). Wenn jedoch nicht

das KOMA-Script-Literaturverzeichnis verwendet wird, dann gibt es eine Warnung aus.

```
\newcommand{\BreakBibliography}[1]{%
  \ClassWarning{\KOMAClassName}{%
    You've used \string\BreakBibliography\space either outside of\MessageBreak
    the bibliography or 'thebibliography' was redefined\MessageBreak
    incompatible, e.g., using a bibliography package.\MessageBreak
    \string\BreakBibliography\space ignored%
  }%
}
```

`\AfterBibliographyPreamble`
`\AtEndBibliography`

Zwei neue Anweisungen, um am Anfang des Literaturverzeichnisses, nach Überschrift und Präambel, und am Ende des Literaturverzeichnisses, nach Abschluss der Liste, Code einschleusen zu können. Das funktioniert natürlich nur so lange, wie keine Paket verwendet wird, das die Umgebung umdefiniert.

```
\newcommand{\AfterBibliographyPreamble}{\g@addto@macro\bib@before@hook}
\newcommand{\AtEndBibliography}{\g@addto@macro\bib@after@hook}
```

Datei o scrkidx.dtx

15 Index

Der Index unterscheidet sich bei KOMA-Script scheinbar nicht wesentlich von dem, was die Standardklassen zu bieten haben. In Wirklichkeit sind die Unterschiede jedoch erheblich.

Briefe haben keinen Index.

15.1 Anweisungen und Einstellungen für die Optionen

Einige Anweisungen und Einstellungen werden bereits benötigt, bevor die Optionen abgearbeitet werden.

`\idx@@heading` Diese Anweisung wird verwendet, um nur die Überschrift zu setzen (nichts weiter!). Sie hat genau ein Argument, nämlich den Text der Überschrift und wird innerhalb von `\idx@heading` verwendet.

```
\newcommand*{\idx@@heading}{%
⟨article⟩   \section*
⟨book | report⟩   \chapter*
}
```

15.2 Optionen für den Index

`index` `\idx@@heading` Die Option sorgt dafür, dass der Index auch im Inhaltsverzeichnis erscheint. Dazu wird das interne Makro `\idx@@heading` umdefiniert, über das die Überschrift gesetzt und der Kolumnentitel angepasst wird. An dieser Stelle muss dafür gesorgt werden, dass im Falle von scrbook und scrreprt das Makro `\@schapter` verwendet wird. Damit dies hier möglich ist mussten tiefreifendere Änderungen am LaTeX-Kernel bzw. am Gliederungsbefehl `\chapter` vorgenommen werden.

```
\KOMA@key{index}{%
  \KOMA@set@ncmdkey{index}{@tempa}{%
    {notoc}{0},{nottotoc}{0},{default}{0},{plainheading}{0},%
    {totoc}{1},{toc}{1},{notnumbered}{1}%
  }{#1}%
  \ifcase \@tempa\relax
    \renewcommand*{\idx@@heading}{%
⟨article⟩     \section*
⟨book | report⟩     \chapter*
    }%
  \or
    \renewcommand*{\idx@@heading}{%
```

```
⟨article⟩      \addsec
⟨book | report⟩      \addchap
      }%
   \fi
}
```

idxtotoc

```
\KOMA@DeclareDeprecatedOption{idxtotoc}{index=totoc}
```

15.3 Definitionen für den Index

\index@preamble
\setindexpreamble

Analog zu \bibpreamble kann \index@preamble vom Benutzer mit \setindexpreamble auf einen Text gesetzt werden, der nach der Indexüberschrift eingefügt wird. Hier wird ein echtes internes Makro verwendet, da nicht auf Kompatibilität geachtet werden muss.

```
\newcommand*{\index@preamble}{}
\let\index@preamble=\relax
\newcommand{\setindexpreamble}[1]{\def\index@preamble{#1}}
```

\idx@heading

Befehl zum Setzen der Überschrift des Indexes. Bis Version 2.7a sah diese Definition so aus:

```
\newcommand*{\idx@heading}{%
  \if@twoside\cleardoublepage\else\clearpage\fi%
  \@mkboth{\indexname}{\indexname}%
  \twocolumn[%
%<article>  \section*%
%<report|book>  \@makeschapterhead%
    {\indexname}\index@preamble]
}
```

Dagegen ist grundsätzlich nichts zu sagen. Allerdings gibt es seit Version 2.7b die neue Methode der Kapitel-Präambel, die extra eingeführt wurde, um die hyperref-Links – insbesondere den des Index – zumindest bei Verwendung von hpdftex.def zu verbessern. Deshalb wurde die Implementierung entsprechend abgeändert. Außerdem wurde das Setzen der Präambel aus Konsistenzgründen in Version 2.7j übereinstimmten mit der thebibliography-Umgebung in die theindex-Umgebung verschoben.

```
\newcommand*{\idx@heading}{%
⟨report | book⟩  \if@openright\cleardoublepage\else\clearpage\fi%
  \twocolumn[\idx@@heading{\indexname}]%
  \@mkboth{\indexname}{\indexname}%
}
```

theindex

Der Index ist wie üblich definiert.

```
\newenvironment{theindex}{%
  \if@twocolumn
    \@restonecolfalse
```

```
  \else
    \@restonecoltrue
  \fi
  \columnseprule \z@
  \columnsep 35\p@
⟨book | report⟩  \setchapterpreamble{\index@preamble}%
  \idx@heading%
⟨article⟩  \index@preamble\par\nobreak
  \thispagestyle{\indexpagestyle}\parindent\z@
  \setlength{\parskip}{\z@ \@plus .3\p@}%
  \setlength{\parfillskip}{\z@ \@plus 1fil}%
  \let\item\@idxitem
}{%
  \if@restonecol\onecolumn\else\clearpage\fi
}
```

`\@idxitem` Hier haben wird den Anfang eines Eintrags in den Index

`\subitem` und ebenso Unter- und Unteruntereinträge sowie ein vertikaler Abstand.

`\subsubitem`
`\indexspace`

```
\newcommand*\@idxitem{\par\hangindent 40\p@}
%    \begin{macrocode}
\newcommand*\subitem{\@idxitem \hspace*{20\p@}}
\newcommand*\subsubitem{\@idxitem \hspace*{30\p@}}
\newcommand*\indexspace{%
  \par \vskip 10\p@ \@plus5\p@ \@minus3\p@\relax
}
```

15.4 Standardbezeichner

`\indexname` Names des Index:

```
\newcommand*\indexname{Index}
```

Datei p scrklist.dtx

16 Listen und Tabellen

KOMA-Script verfügt über einige zusätzliche Listenumgebungen. Ein Teil dieser Listenumgebungen ist nicht sofort auf den ersten Blick als solche zu erkennen, so wie dies schon bei den Standardklassen beispielsweise mit `quote` und `quotation` der Fall ist.

16.1 Optionen

Diese Umgebungen werden nicht durch Optionen beeinflusst.

16.2 Ränder in Listen

`leftmargini` `leftmarginii` `leftmarginiii` `leftmarginiv` `leftmarginv` `leftmarginvi` `leftmargin` `labelsep` `labelwidth`

Für die Listenumgebungen in verschiedenen Stufen müssen Ränder definiert werden. Hinzu kommen noch die Breite eines Labels und der Abstand nach einem Label.

```
⟨*!letter⟩
\if@twocolumn
  \setlength{\leftmargini}{2em}
\else
⟨/!letter⟩
  \setlength{\leftmargini}{2.5em}
⟨!letter⟩\fi
\setlength{\leftmarginii}{2.2em}
\setlength{\leftmarginiii}{1.87em}
\setlength{\leftmarginiv}{1.7em}
⟨*!letter⟩
\if@twocolumn
  \setlength{\leftmarginv}{.5em}
  \setlength{\leftmarginvi}{.5em}
\else
⟨/!letter⟩
  \setlength{\leftmarginv}{1em}
  \setlength{\leftmarginvi}{1em}
⟨!letter⟩\fi
\setlength{\leftmargin}{\leftmargini}
\setlength{\labelsep}{.5em}
\setlength{\labelwidth}{\leftmargini}
\addtolength{\labelwidth}{-\labelsep}
```

Weil die Gelegenheit so günstig ist, setzen wir hier auch gleich noch ein paar *penalties* (Erlaubsnispunkte).

```
\@beginparpenalty=-\@lowpenalty
\@endparpenalty  =-\@lowpenalty
\@itempenalty    =-\@lowpenalty
```

16.3 Verschiedene Längen für Tabellen, Arrays und Tabulatoren

Vorbemerkung: Arrays sind im Prinzip Tabellen zur Verwendung im mathemathischen Modus.

arraycolsep
tabcolsep
arrayrulewidth
doublerulewidth
tabbingsep

Bei diesen Längen handelt es sich um die Abstände der Spalten in `array`- und `tabular`-Umgebungen, sowie um die Breite und den Abstand der Trennlinien. Hinzu kommt noch der `\'` Abstand in `tabbing`-Umgebungen.

```
\setlength\arraycolsep{5\p@}
\setlength\tabcolsep{6\p@}
\setlength\arrayrulewidth{.4\p@}
\setlength\doublerulesep{2\p@}
\setlength\tabbingsep{\labelsep}
```

16.4 Marken in Listen

\theenumi
\theenumii
\theenumiii
\theenumiv
\labelenumi
\labelenumii
\labelenumiii
\labelenumiv
\p@enumii
\p@enumiii
\p@enumiv
\labelitemi
\labelitemii
\labelitemiii
\labelitemiv

Für die numerierten Listenumgebungen müssen Marken bzw. Numerierungen festgelegt werden. Dazu wird eine Darstellung der Zähler, ihr Prefix und ihre Labeldarstellung neu definiert.

```
\renewcommand*\theenumi{\@arabic\c@enumi}
\renewcommand*\theenumii{\@alph\c@enumii}
\renewcommand*\theenumiii{\@roman\c@enumiii}
\renewcommand*\theenumiv{\@Alph\c@enumiv}
\newcommand*\labelenumi{\theenumi.}
\newcommand*\labelenumii{\theenumii)}
\newcommand*\labelenumiii{\theenumiii.}
\newcommand*\labelenumiv{\theenumiv.}
\renewcommand*\p@enumii{\theenumi}
\renewcommand*\p@enumiii{\p@enumii(\theenumii)}
\renewcommand*\p@enumiv{\p@enumiii\theenumiii}
\newcommand*\labelitemi{\textbullet}
\newcommand*\labelitemii{\normalfont\bfseries\textendash}
\newcommand*\labelitemiii{\textasteriskcentered}
\newcommand*\labelitemiv{\textperiodcentered}
```

16.5 Definition der Umgebungen

description

Die `description`-Umgebung dient der Beschreibung von einzelnen Begriffen. Der Begriff aus dem optionalen Argument des `\item`-Befehls wird in einem speziellen Font, dem `\descfont`, gesetzte.

```
\newenvironment{description}{%
  \list{}{\labelwidth\z@ \itemindent-\leftmargin
    \let\makelabel\descriptionlabel}%
```

```
}{%
  \endlist
}
\newcommand*{\descriptionlabel}[1]{%
  \hspace{\labelsep}\descfont #1%
}
```

`labeling`
`labelinglabel`

Die `\labeling`-Umgebung ist eine Erweiterung des Script Pakets. Sie erwartet ein optionales und ein normales Argument. Das optionale Argument beschreibt einen speziellen Trenntext zwischen `\item`-Marke und `\item`-Beschreibung. Das eigentliche Argument wird für die Ermittlung der Einrückung des Beschreibungstextes bzw. des Trenntextes benötigt.

```
⟨/class⟩
\newenvironment{labeling}[2][]{%
  \def\sc@septext{#1}%
  \list{}{\settowidth{\labelwidth}{{%
        \usekomafont{labelinglabel}{#2%
          \usekomafont{labelingseparator}{\sc@septext}}%
      }}%
    \leftmargin\labelwidth \advance\leftmargin by \labelsep
    \let\makelabel\labelinglabel
  }%
}{%
  \endlist
}
\newcommand\labelinglabel[1]{%
  \usekomafont{labelinglabel}{#1\hfil
    \usekomafont{labelingseparator}{\sc@septext}}%
}
⟨*class⟩
```

`verse`

Die Verse-Umgebung ist für Zitate in Gedichtform und ähnliches gedacht.

```
\newenvironment{verse}{%
  \let\\=\@centercr
  \list{}{\itemsep=\z@
    \itemindent=-1.5em
    \listparindent=\itemindent
    \rightmargin=\leftmargin
    \advance\leftmargin by1.5em
  }%
  \item\relax
}{%
  \endlist
}
```

`quotation`
`quote`

Die `quotation`- und die `quote`-Umgebung erlauben rechts und links eingerückte Passagen. Absätze werden entweder in der ersten Zeile zusätzlich eingerückt oder durch vertikalen Abstand markiert.

```
\newenvironment{quotation}{%
  \list{}{\listparindent 1em%
    \itemindent    \listparindent
    \rightmargin   \leftmargin
    \parsep        \z@ \@plus\p@
  }%
  \item\relax
}{%
  \endlist
}
\newenvironment{quote}{%
  \list{}{\rightmargin\leftmargin}%
  \item\relax
}{%
  \endlist
}
```

`addmargin`
`addmargin*`

Diese beiden Umgebungen ähneln `quote` und `quotation`. Dabei werden Absatzeinzug und Absatzabstand nicht verändert. Die einzige Veränderung besteht in den Rändern. Wie stark die Ränder verändert werden, hängt dabei von den Parametern ab. Ist nur der obligatorische Parameter angegeben, werden die Ränder auf beiden Seiten um diesen Wert vergrößert. Ist ein optionaler Parameter angegeben, so ist dies bei `addmargin` der linke und bei `addmargin*` der innere Rand. Der obligatorische Parameter ist dann der andere Rand. Diese Entscheidung ist das einzige, was die Umgebungen selbst erledigen müssen.

`\@addmargin`

Der Rest wird von diesem Makro erledigt.

```
⟨/class⟩
\newenvironment{addmargin}{%
  \@tempswafalse\@addmargin
}{%
  \advance\@listdepth\@ne
  \endlist
}
\newenvironment{addmargin*}{%
  \@tempswafalse
  \if@twoside\ifthispageodd{}{\@tempswatrue}\fi
  \@addmargin
}{%
  \advance\@listdepth\@ne
  \endlist
}
\newcommand*{\@addmargin}[2][\@tempa]{%
  \list{}{%
    \if@tempswa
      \def\@tempa{\leftmargin}%
      \setlength{\leftmargin}{#2}%
      \setlength{\rightmargin}{#1}%
```

```
    \else
      \def\@tempa{\rightmargin}%
      \setlength{\rightmargin}{#2}%
      \setlength{\leftmargin}{#1}%
    \fi
    \setlength{\listparindent}{\parindent}%
    \setlength{\itemsep}{\parskip}%
    \setlength{\itemindent}{\z@}%
    \setlength{\topsep}{\z@}%
    \setlength{\parsep}{\parskip}%
    \setlength{\partopsep}{\z@}%
    \let\makelabel\@gobble
    \setlength{\labelwidth}{\z@}%
    \advance\@listdepth\m@ne
  }%
  \item\relax%
}
⟨*class⟩
```

16.6 Schriftarten für Listen

`\descfont` Dies ist die Schriftart, in der das Label der Einträge in eine `description`-Umgebung gesetzt wird. Das Makro ist als intern zu betrachten. Anwender habe stattdessen das entsprechende Element zu verwenden.

```
\newcommand*\descfont{\sffamily\bfseries}
```

`\scr@fnt@descriptionlabel` Das Element auf das `\descfont` angewandt wird:

```
\newcommand*{\scr@fnt@descriptionlabel}{\descfont}
```

`labelinglabel` `labelingseparator` Das Label einer `labeling`-Umgebung wird in eigener Schrift gesetzt. Innerhalb des Labels wird dann das Trennzeichen ebenfalls mit eigener Schrift gesetzt. Voreingestellt ist in beiden Fällen keine Änderung.

```
⟨/class⟩
\newkomafont{labelinglabel}{}
\newkomafont{labelingseparator}{}
⟨*class⟩
```

Datei q
scrlfile.dtx

17 Hooks loading files, packages or classes[2]

The introduction of LaTeX 2_ε in 1994 brought many changes in the handling with LaTeX-extensions. Today the package author has many macros in order to determine if another package or class is employed and whether specific options are used. The author can load other packages or can specify options in the the case the package is loaded later. This has led to the expectation that the order of package-loading will be not important. But this hope has not been fulfilled, since often different packages define or redefine one macro again and again. In such a case the order of package-loading becomes very important. For the user it is sometimes difficult to understand the behaviour and in some cases the user wants only react on the loading of a package. This is also not really simple.

Assuming the simple example that loads the package longtable with a KOMA-Script document-class employed. The longtable package defines table captions suitable for the standard classes, but the captions are totally unsuitable for documents using KOMA-Script and thus the provided configuration commands have no influence. In order to solve this problem the commands which are responsible for the table captions of the longtable package have to be re-defined. But at the moment when the longtable is loaded the KOMA-Script class is already processed.

The only chance for KOMA-Script was to delay the re-definition until the begin of the document with help of the macro `\AtBeginDocument`. If the user wants to change the definitions too, it is recommended to do this in the preamble of the document. But this is impossible since later at `\begin{document}` KOMA-Scriptwill again overwrite the user-definition with its own. Therefore the user has to delay his definition with `\AtBeginDocument` as well.

However, KOMA-Script shouldn't delay the re-definition until `\begin{document}`. It would be enough to delay until the package longtable has been loaded. But unfortunately the basic LaTeX does not define appropriate commands. The package scrlfile provides redress here.

Likewise, it might be conceivable that before a package is loaded one would like to save the definition of a macro in a help-macro, in order to restore its meaning after the package has been loaded. The package scrlfile allows this too.

[2] You'll find a short version of scrlfile documentation here. A longer version may be found at english and german KOMA-Script manual.

The employment of scrlfile is not limited to package dependencies only. Even dependencies with any other file can be attended. For example the user can be warned if the not uncritical file `french.ldf` has been loaded.

Though the package is particularly interesting for package authors, there are of course applications for normal LaTeX users too. Therefore this chapter gives and explains examples for both groups of users.

`\BeforeFile` `\AfterFile`

The command `\BeforeFile{`⟨*file*⟩`}{`⟨*instructions*⟩`}` enables to execute ⟨*instructions*⟩ the next time the ⟨*file*⟩ is loaded. In the same way works `\AfterFile`, but here the ⟨*instructions*⟩ will be executed after the ⟨*file*⟩ has been loaded. If ⟨*file*⟩ will never be loaded then the ⟨*instructions*⟩ will never be executed.

In order to implement those features scrlfile re-defines the well known LaTeX command `\InputIfFileExists`. If this macro has not the expected definition scrlfile gives a warning. This is for the case that in future LaTeX versions the macro can have a different definition or an other package has already re-defined it.

The command `\InputIfFileExists` is used everytime LaTeX has to load a file. This is independent from whether the actual command was `\LoadClass`, `\documentclass`, `\usepackage`, `\RequiresPackage`, or `\include`. Exclusively the command

```
\input foo
```

loads the file ⟨*foo*⟩ without to utilize `\InputIfFileExists`. Therefore one should always use

```
\input{foo}
```

instead. Notice the parentheses surrounding the file name!

`\BeforeClass` `\BeforePackage`

The command `\BeforeClass{`⟨*class*⟩`}{`⟨*instructions*⟩`}` and the command `\BeforePackage{`⟨*package*⟩`}{`⟨*instructions*⟩`}` work the same way like the above described `\BeforeFile`. The only difference is that the document ⟨*class*⟩ and the LaTeX ⟨*package*⟩ are specified with their names and not with their file names. That means the file extensions "`.cls`" and "`.sty`" have to be omitted.

`\AfterClass` `\AfterClass*` `\AfterPackage` `\AfterPackage*`

The command `\AfterClass{`⟨*class*⟩`}{`⟨*instructions*⟩`}` and the command `\AfterPackage{`⟨*package*⟩`}{`⟨*instructions*⟩`}` work the same way like command `\AfterFile`. The only difference is that the document ⟨*class*⟩ and the LaTeX ⟨*package*⟩ are specified with their names and not with their file names. That means the file extensions "`.cls`" and "`.sty`" have to be omitted. The starred versions execute the ⟨*instructions*⟩ not only next time the class or package is loaded, but also immediately if the class or package has been loaded already.

`\ReplaceInput`

The command `\ReplaceInput{`⟨*name of file to be replaced*⟩`}{`⟨*name of replacement file*⟩`}` sets up an replacement. Whenever LaTeX is told to input the file with ⟨*name of file to be replaced*⟩ it will input the file with ⟨*name*

of replacement file⟩ instead. With this you may even teill LaTeXto input e.g. "`\jobname.xua`" instead of "`\jobname.aux`" using:

```
\ReplaceInput{\jobname.aux}{\jobname.xua}
```

If you replace a replacement file both, the first replaced file and the replacement file will be replaced by the new replacement file. So

```
\ReplaceInput{\jobname.aux}{\jobname.xua}
\ReplaceInput{\jobname.xua}{\jobname.uxa}
```

will replace "`\jobname.aux`" and "`\jobname.xua`" by "`\jobname.uxa`". But ring replacement like

```
\ReplaceInput{\jobname.aux}{\jobname.xua}
\ReplaceInput{\jobname.xua}{\jobname.aux}
```

would result in an stack size error. So you cannot input an already replaced file using LaTeX commands.

`\ReplaceClass`
`\ReplacePackage`

You should not replace a class or package using `\ReplaceInput`. If you do so, you'll get errors because of not matching class names. But you may use `\ReplaceClass{`⟨*name of class to be replaced*⟩`}{`⟨*name of replacement class*⟩`}` and `\ReplacePackage{`⟨*name of package to be replaced*⟩`}{`⟨*name of replacement package*⟩`}` instead. With this commands like `\documentclass{`⟨*name of class to be replaced*⟩`}` would work like `\documentclass{`⟨*name of replacement class*⟩`}`. Effected commands are `\LoadClass`, `\LoadClassWithOptions`, `\documentclass`, `\RequirePackage`, `\RequirePackageWithOptions`, `\usepackage`.

Note: The replacement class or package will be loaded with the same option list the replaced class or package would have been. If you replace a class or package with a class or package that does not know the options of the replaced class or package, you may declare the missing options using `\BeforeClass` and `\BeforePackage`.

If you e.g. want to load package `newfoo` instead of package `oldfoo` you may use:

```
\ReplacePackage{oldfoo}{newfoo}
```

And if the old package had an option `oldopt` but the new package does not you may use

```
\BeforePackage{newfoo}{%
  \DeclareOption{oldopt}{%
    \PackageInfo{newfoo}{option ``oldopt'' not supported}%
  }%
}
```

If new option `newopt` should be used instead of `oldopt` you may use:

```
\BeforePackage{newfoo}{%
  \DeclareOption{oldopt}{%
    \ExecuteOption{newopt}%
  }%
}
```

You may also set new default options using:

```
\BeforePackage{newfoo}{%
  \DeclareOption{oldopt}{%
    \ExecuteOption{newopt}%
  }%
  \PassOptionsToPackage{newdefoptA,newdefoptB}{newfoo}%
}
```

or simply

```
\BeforePackage{newfoo}{%
  \DeclareOption{oldopt}{%
    \ExecuteOption{newopt}%
  }%
}
\PassOptionsToPackage{newdefoptA,newdefoptB}{newfoo}%
```

Note: To replace classes you have to load scrlfile using `\RequirePackage` before loading the class.

`\PreventPackageFromLoading`
`\ResetPreventPackageFromLoading`
`\StorePreventPackageFromLoading`

The command `\PreventPackageFromLoading{`⟨*package list*⟩`}` does not replace a package by another but prevents packages from beeing loaded even if they should be loaded using `\RequirePackage` or `\usepackage`. The ⟨*package list*⟩ is a comma seperated list of packages, that should never be loaded. If you use the command several times, the later ⟨*package lists*⟩ will be added to the already existing. You may use `\ResetPreventPackageFromLoading` to clean the ⟨*package list*⟩. If you need to know the packages at the ⟨*package list*⟩, you may use `\StorePackageFromLoading{`⟨
to store the current list at macro ⟨*command*⟩.

`\BeforeClosingMainAux`

The command `\BeforeClosingMainAux{`⟨*instructions*⟩`}` works different from all the commands above. Is isn't a load-file-hook. Package authors often have the problem, that they need to write something to `aux`-file after output of the last page. This is often done with:

```
\AtEndDocument{\if@filesw
    \write\@auxout{%
      Write this to aux.
    }%
  \fi}
```

But this fails, if the last `\shipout` was already done e.g. because of someone used a different solution:

```
\AtEndDocument{\if@filesw\clearpage
    \immediate\write\@auxout{%
      Write this to aux.
    }%
  \fi}
```

The best solution would be to immediate write to aux-file just after the \clearpage and before closing aux-file at \end{document}. That's the goal of \BeforeClosingMainAux:

```
\BeforeClosingMainAux{\if@filesw
    \immediate\write\@auxout{%
      Write this to aux.
    }%
  \fi}
```

This even works, if \clearpage at \end{document} will not result in \shipout or if typessetting code — even \clearpage — has been used at \AtEndDocument.

The \BeforeClosingMainAux hook should not be used for code that does any typesetting since the result of \BeforeClosingMainAux would be unpredictable.

\AfterReadingMainAux

The command \AfterReadingfMainAux{⟨*instructions*⟩} works similar to \BeforeClosingMainAux, but instead of processing instructions before closing the main aux-file, they will be processed after reading the main aux-file at \begin{document}.

The \AfterReadingMainAux hook should not be used for code that does any typesetting since the result of \AfterReadingMainAux would be unprecitable.

17.1 Preamble

First of all we show the package and version at terminal and log-file:

```
\typeout{Package scrlfile, \csname ver@scrlfile.sty\endcsname%
  ^^J\@spaces\@spaces\@spaces\@spaces\space\space
  Copyright (C) Markus Kohm^^J}
```

17.2 Options

Currently there are no options.

```
\ExecuteOptions{}
\ProcessOptions\relax
```

17.3 The macros

\InputIfFileExists

LaTeX uses this macro for loading files with \input{⟨*filename*⟩}, \include, \LoadClass, \LoadClassWithOptions, \documentclass, \documentstyle, \RequirePackage, \usepackage. Only \input ⟨*filename*⟩ loads ⟨*filename*⟩ without using \InputIfFileExists.

Because of this, it's very easy to add stuff before and after loading a file. We simply change this macro and add to macros — one before loading the file and one after loading the file. But before we redefine the macro, we test, if the macro was defined like we expected. LaTeX will warn, if not.

```
\CheckCommand{\InputIfFileExists}[2]{%
  \IfFileExists{#1}%
    {#2\@addtofilelist{#1}\@@input \@filef@und}}
```

Now we can redefine the macro like described above:

```
\renewcommand{\InputIfFileExists}[2]{%
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname #1-@alias\endcsname\relax
    \expandafter\@secondoftwo
  \else
    \scr@replacefile@msg{\csname #1-@alias\endcsname}{#1}%
    \expandafter\@firstoftwo
  \fi
  {%
    \expandafter\InputIfFileExists\expandafter{\csname
      #1-@alias\endcsname}{#2}%
  }%
  {\IfFileExists{#1}{%
      \scr@load@hook{before}{#1}%
      #2\@addtofilelist{#1}%
      \@@input \@filef@und
      \scr@load@hook{after}{#1}%
    }}%
}
```

\@loadwithoptions
\scr@loadwithoptions
\scr@@loadwithoptions

LaTeX uses this macro for loading classes and packages using \LoadClassWithOptions or \RequirePackageWithOptions. We simply say now: if the class or package should be replaced by another, try it again with the replacement name. The stack is used. This will result in an error if the replacements are nested too deep. This should be, because it will also result in an error using ring replacements.

```
\newcommand*{\scr@@loadwithoptions}{}
\newcommand*{\scr@loadwithoptions}{}
\let\scr@loadwithoptions\@loadwithoptions
\renewcommand*{\@loadwithoptions}[3]{%
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname #3.#1-@aliasname\endcsname\relax
    \def\scr@@loadwithoptions{\scr@loadwithoptions{#1}{#2}{#3}}%
```

```
  \else
    \scr@replacefile@msg{\csname #3.#1-@aliasname\endcsname.#1}{#3.#1}%
    \def\scr@@loadwithoptions{%
      \@loadwithoptions{#1}{#2}{\csname #3.#1-@aliasname\endcsname}%
    }%
  \fi
  \scr@@loadwithoptions
}
```

\@onefilewithoption
\scr@onefilewithoption
\scr@@onefilewithoption

LaTeX uses this macro for loading classes and packages using \LoadClass, \documentclass, \RequirePackage, and \usepackage. We simply say now: if the class or package should be replaced by another, try it again with the replacement name. The stack is used. This will result in an error if the replacements are nested too deep. This should be, because it will also result in an error using ring replacements.

```
\newcommand*{\scr@onefilewithoptions}{}
\let\scr@onefilewithoptions\@onefilewithoptions
\def\@onefilewithoptions#1[#2][#3]#4{%
  \begingroup\expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname #1.#4-@aliasname\endcsname\relax
```

Do this ridiculous definition and call outside of \ifx... \fi because of some errornous packages like some versions of breakurl. This does not avoid an error, but the message about unclosed \if will be outside of scrlfile.

```
    \def\scr@@onefilewithoptions{\scr@onefilewithoptions{#1}}%
  \else
    \scr@replacefile@msg{\csname #1.#4-@aliasname\endcsname.#4}{#1.#4}%
    \edef\scr@@onefilewithoptions{%
      \noexpand\@onefilewithoptions{\csname #1.#4-@aliasname\endcsname}}%
  \fi
  \scr@@onefilewithoptions[{#2}][{#3}]{#4}%
}
```

\ReplaceInput

Defining the replacement by definition of \#1-@alias.

```
\newcommand*{\ReplaceFile}[2]{%
  \expandafter\edef\csname #1-@alias\endcsname{#2}%
}
```

\ReplacePackage
\ReplaceClass
\@replacefilewithext

Note that replacing a package or class will not replace the input of the file of the package or class but only the input of package or class.

```
\newcommand*{\@replacefilewithext}[3]{%
  \expandafter\edef\csname #1.#3-@aliasname\endcsname{#2}%
}
\newcommand*{\ReplacePackage}[2]{%
  \@replacefilewithext{#1}{#2}\@pkgextension
}
\newcommand*{\ReplaceClass}[2]{%
  \@replacefilewithext{#1}{#2}\@clsextension
}
```

\scr@replacefile@msg

Last but not least scrlfile will write a message to the log file at every replacement.

```
\newcommand*{\scr@replacefile@msg}[2]{%
  \PackageInfo{scrlfile}{inputing ‘#1’ instead of ‘#2’}%
}
```

\scr@load@hook

This macro simply calls a load-file-hook, either the *before* or the *after* one, if the hook exists. After calling the hook it will be deleted. First argument is the type of hook (⟨*before*⟩ or ⟨*after*⟩), second argument is a filename.

```
\newcommand*{\scr@load@hook}[2]{%
  \@ifundefined{#2-@#1}{}{%
    \@nameuse{#2-@#1}%
    \expandafter\let\csname #2-@#1\endcsname=\relax}}
```

\scr@set@load@hook

This command adds stuff to a load-file-hook, either the *before* or the *after* one. First argument is the type of hook (⟨*before*⟩ or ⟨*after*⟩), second is a filename. There's a third argument not parsed by the macro itself but \g@addto@macro. This third argument is the stuff, which will be added to the hook. Because of using \g@addto@macro the hook must be defined. So we define it, if it isn't.

```
\newcommand*{\scr@set@load@hook}[2]{%
  \@ifundefined{#2-@#1}{\@namedef{#2-@#1}{}}{}%
  \expandafter\g@addto@macro\csname #2-@#1\endcsname}
```

\BeforeFile
\AfterFile
\BeforeClass
\AfterClass
\@AfterClass
\@sAfterClass
\BeforePackage
\AfterPackage
\@AfterPackage
\@sAfterPackage

These macros add stuff to a load-file-hook of a file, class or package. First argument is the name of the file (with extension), class (without extension) or package (without extension). There's a second argument parsed by wether the macros themself nor \scr@set@load@hook but \g@addto@macro. See \scr@set@load@hook for more information. The star-forms \AfterClass* and \AfterPackage* does the action immediate, if the class or package was already loaded.

```
\newcommand*{\BeforeFile}{\scr@set@load@hook{before}}
\newcommand*{\AfterFile}{\scr@set@load@hook{after}}
\newcommand*{\BeforeClass}[1]{%
  \scr@set@load@hook{before}{#1.\@clsextension}}
\newcommand*{\AfterClass}{%
  \@ifstar {\@sAfterClass}{\@AfterClass}}
\newcommand*{\@AfterClass}[1]{%
  \scr@set@load@hook{after}{#1.\@clsextension}}
\newcommand*{\@sAfterClass}[1]{%
  \begingroup
    \@ifclassloaded{#1}{%
      \aftergroup\@secondoftwo
    }{%
      \aftergroup\@AfterClass
    }%
  \endgroup
```

```
  {#1}%
}
\newcommand*{\BeforePackage}[1]{%
  \scr@set@load@hook{before}{#1.\@pkgextension}}
\newcommand*{\AfterPackage}{%
  \@ifstar {\@sAfterPackage}{\@AfterPackage}}
\newcommand*{\@AfterPackage}[1]{%
  \scr@set@load@hook{after}{#1.\@pkgextension}}
\newcommand*{\@sAfterPackage}[1]{%
  \begingroup
    \@ifpackageloaded{#1}{%
      \aftergroup\@secondoftwo
    }{%
      \aftergroup\@AfterPackage
    }%
  \endgroup
  {#1}%
}
```

\PreventPackageFromLoading
\scrlfile@excludedpackages
\scrlfile@saved@RequirePackage

```
\newcommand*{\scrlfile@excludedpackages}{}
\let\scrlfile@excludedpackages\@empty
\newcommand*{\scrlfile@saved@RequirePackage}{}
\newcommand*{\PreventPackageFromLoading}[1]{%
```

We do nothing, if the argument was empty:

```
  \edef\reserved@b{#1}%
  \ifx \reserved@b\@empty
    \PackageWarning{scrlfile}{%
      \string\PreventPackageFromLoading\space with empty argument\MesageBreak
      ignored%
    }%
  \else
```

If the argument was not empty we first need to patch `\RequirePackage` and maybe `\usepackage` too.

```
    \ifx\scrlfile@excludedpackages\@empty
      \let\scrlfile@saved@RequirePackage\RequirePackage
      \let\RequirePackage\scrlfile@RequirePackage
      \def\reserved@a#1{%
        \@latex@error
        {\noexpand \usepackage before \string\documentclass}%
        {\noexpand \usepackage may only appear in the document
          preamble, i.e.,\MessageBreak
          between \noexpand\documentclass and
          \string\begin{document}.}%
        \@gobble}%
      \ifx\reserved@a\usepackage\else
        \ifx\usepackage\scrlfile@saved@RequirePackage\else
          \PackageWarning{scrlfile}{%
```

```
            Maybe serious problem: unexpected definition of\MessageBreak
            \string\usepackage
          }%
        \fi
        \let\usepackage\RequirePackage
      \fi
    \fi
```

We have to test, if a package was already loaded. Only not yet loaded packages will be added to the exclusion list. Empty arguments will be ignored.

```
    \expandafter\@for \expandafter\reserved@a\expandafter:\expandafter=%
    \reserved@b\do {%
      \ifx\reserved@a\@empty\else
        \expandafter\@ifpackageloaded\expandafter{\reserved@a}{%
          \PackageWarning{scrlfile}{%
            package `\reserved@a' already loaded.\MessageBreak
            Cannot prevent it from beeing loaded%
          }%
        }{%
          \edef\scrlfile@excludedpackages{,\reserved@a}%
        }%
      \fi
    }%
```

If the exclusion list is still empty (because all packages has been loaded already), remove the patch:

```
    \ifx\scrlfile@excludedpackages\@empty
      \ResetPreventPackageFromLoading
    \fi
  \fi
}
\@onlypreamble\PreventPackageFromLoading
```

`ResetPreventPackageFromLoading`
`StorePreventPackageFromLoading`

These two are very simple

```
\newcommand*{\ResetPreventPackageFromLoading}{%
  \let\scrlfile@excludedpackages\@empty
  \ifx\RequirePackage\scrlfile@RequirePackage
    \ifx\usepackage\RequirePackage
      \let\usepackage\scrlfile@saved@RequirePackage
    \fi
    \let\RequirePackage\scrlfile@saved@RequirePackage
  \fi
}
\@onlypreamble\ResetPreventPackageFromLoading
\newcommand*{\StorePreventPackageFromLoading}[1]{%
  \let#1\scrlfile@excludedpackages
}
\@onlypreamble\StorePreventPackageFromLoading
```

\scrlfile@RequirePackage This macro will replace \RequirePackage whenever the exlude list is not empty

```
\newcommand*{\scrlfile@RequirePackage}[2][]{%
  \let\reserved@c\@empty
  \@for\reserved@b:=#2\do {%
    \begingroup
      \@tempswatrue
      \@for\reserved@a:=\scrlfile@excludedpackages\do {%
        \ifx\reserved@a\reserved@b
          \PackageInfo{scrlfile}{Excluding package `\reserved@b'\MessageBreak
            from loading}%
          \@tempswafalse
        \fi
      }%
      \if@tempswa
        \xdef\reserved@c{\reserved@c,\reserved@b}%
      \fi
    \endgroup
  }%
  \ifx\reserved@c\@empty
    \let\reserved@a\scr@gobbleopt
  \else
    \edef\reserved@a{\noexpand\scrlfile@saved@RequirePackage[#1]{%
        \expandafter\@gobble\reserved@c\@empty}}%
  \fi
  \reserved@a
}
```

\scr@gobbleopt Eat the optional argument if it exists.

```
\newcommand{\scr@gobbleopt}[1][]{}
```

\BeforeClosingMainAux
\AfterReadingMainAux These are completly different. First we have to establish a new hook at \end{document}, just after closing the group and after the final \clearpage.

```
\AtEndDocument{%
  \let\scr@saved@checkend=\@checkend
  \renewcommand*{\@checkend}[1]{%
    \def\reserved@a{#1}\def\reserved@b{document}%
    \scr@saved@checkend{#1}%
    \ifx\reserved@a\reserved@b
      \let\scr@saved@clearpage=\clearpage
      \renewcommand*{\clearpage}{\scr@saved@clearpage
        \scr@hook@bcma
        \scr@restore@newpage
        \let\clearpage=\scr@saved@clearpage}%
    \fi%
    \let\scr@saved@dofilelist\@dofilelist
    \def\@dofilelist{%
      \scr@hook@acma
```

```
      \let\@dofilelist\scr@saved@dofilelist
      \scr@saved@dofilelist
    }%
  }%
}
```

\scr@saved@checkend
\scr@saved@dofilelist

You may see above, that we need save-to macros.

```
\newcommand*{\scr@saved@checkend}{}
\newcommand*{\scr@saved@dofilelist}{}
```

\scr@redefine@newpage

Some classes (e.g. letter) write output to the aux file. Because of this, redefinition of \newpage will only be done, if \BeforeClosingMainAux is used.

\scr@saved@newpage

And \newpage will be restored after the hook.

\scr@restore@newpage

```
\newcommand*{\scr@redefine@newpage}{%
  \let\scr@saved@newpage\newpage
  \def\scr@restore@newpage{%
    \renewcommand*{\newpage}{%
      \PackageWarningNoLine{%
        scrlfile%
      }{%
        \string\newpage\space at main aux file!\MessageBreak
        You have output material at the main aux file.\MessageBreak
        Note, that the \string\BeforeClosingMainAux\space commands\MessageBreak
        are processed before the output from aux file%
      }%
      \let\newpage\scr@saved@newpage
      \newpage
    }%
  }%
  \renewcommand*{\newpage}{%
    \PackageError{%
      scrlfile%
    }{%
      \string\newpage\space at \string\BeforeClosingMainAux
    }{%
      You should never use \string\newpage\space or
      \string\clear(double)page or other output material at the\MessageBreak
      argument of \string\BeforeClosingMainAux!\MessageBreak
      If you'll continue, the \string\newpage\space will be ignored to
      avoid corruption of\MessageBreak
      every prior use of \string\BeforeClosingMainAux.
    }%
  }%
}
\newcommand*{\scr@restore@newpage}{}
```

\scr@hook@bcma
\scr@hook@acma

We also need macros, that saves the commands at the hooks and are extendable using \g@addto@macro.

```
\newcommand*{\scr@hook@bcma}{}
\newcommand*{\scr@hook@acma}{}
```

Now we are able to write the user command. Here we also warn, if we detect one more \newpage — maybe at a \clearpage or \cleardoublepage.

```
\newcommand*{\BeforeClosingMainAux}{%
  \ifx\scr@hook@bcma\@empty
    \gdef\scr@hook@bcma{%
      \renewcommand*{\BeforeClosingMainAux}[1]{####1}%
      \let\protected@write\protected@immediate@write
      \scr@redefine@newpage
    }%
  \fi
  \g@addto@macro\scr@hook@bcma%
}
\newcommand*{\AfterReadingMainAux}{%
  \ifx\scr@hook@acma\@empty
    \gdef\scr@hook@acma{%
      \renewcommand*{\AfterReadingMainAux}[1]{####1}%
      \let\protected@write\protected@immediate@write
    }%
  \fi
  \g@addto@macro\scr@hook@acma%
}
```

\protected@immediate@write

Like LaTeX kernel's \protected@write but using \immediate\write. In this case it is even not a good idea to protect \thepage!

```
\long\def\protected@immediate@write#1#2#3{%
  \begingroup
    #2%
    \let\protect\@unexpandable@protect
    \edef\reserved@a{\immediate\write#1{#3}}%
    \reserved@a
  \endgroup
  \if@nobreak\ifvmode\nobreak\fi\fi
}
```

File r
scrlogo.dtx

18 KOMA-Script Logo

\KOMAScript Das KOMA-Script-Logo wird in allen KOMA-Script-Paketen und -Klassen definiert, falls es nicht bereits definiert ist. Dabei werden die Versalien moderat gesperrt. Es wird jedoch darauf verzichtet, die Versalien etwa einen Punkt kleiner zu setzen, da das Logo aktiv ausgezeichnet erscheinen soll.

```
\@ifundefined{KOMAScript}{%
  \DeclareRobustCommand{\KOMAScript}{\textsf{K\kern.05em O\kern.05em%
      M\kern.05em A\kern.1em-\kern.1em Script}}}{}
```

Datei s scrkcomp.dtx

19 Kompatibilität zu früheren Versionen

Manchmal ist es sinnvoll, dass sich eine neue Version von KOMA-Script etwas anders verhält als frühere Versionen. Gleichzeitig ist es aber für den Anwender manchmal auch notwendig, dass sich neue Versionen ganz genau so verhalten wie frühere. Daher wird eine Option geboten, mit der man die Kompatibilität selbst steuern kann. Voreingestellt ist jeweils maximale Kompatibilität.

19.1 Option

Die gesamte Kompatibilitätssteuerung erfolgt mit einer einzigen Option, bei der man angibt, zu welcher Version Kompatibilität hergestellt werden soll. Dies bedeutet ggf. dann auch, dass einzelne neuere Möglichkeiten nicht zur Verfügung stehen.

`version`
`\scr@compatibility`
`\scr@ta@compatibility`

In einigen Fällen sind Verbesserungen nicht kompatibel mit früheren Versionen. Deshalb sind solche Verbesserungen nur verfügbar, wenn mit diesem Schalter die neue Version ausgewählt wird. Aber es gilt: Entweder kompatibel oder in allen Dingen neu. Mischmasch machen wir nicht. Die aktuell eingestellte Kompatibilität wird in `\scr@compatibility` als Zahl gespeichert. In den Makros `\scr@v@`*`Version`* werden die zugehörigen Nummern gespeichert.

```
⟨class⟩\newcommand*
⟨package⟩\providecommand*
  {\scr@compatibility}{\scr@v@last}
⟨typearea⟩\newcommand*{\scr@ta@compatibility}{\scr@compatibility}
\KOMA@key{version}[last]{%
  \scr@ifundefinedorrelax{scr@v@#1}{%
    \def\scr@compatibility{0}%
⟨class⟩     \ClassWarningNoLine{\KOMAClassName}{%
⟨package⟩     \PackageWarningNoLine{%
⟨extend⟩       scrextend%
⟨typearea⟩       typearea%
⟨package⟩     }{%
      You have set option ‘version’ to value ‘#1’, but\MessageBreak
      this value of version is not supported.\MessageBreak
      Because of this, version was set to ‘first’%
    }%
  }{%
⟨class⟩     \ClassInfoNoLine{\KOMAClassName}{%
⟨package⟩     \PackageInfoNoLine{scrextend}{%
```

```
      Switching compatibility level to ‘#1’%
    }%
⟨class | extend⟩     \edef\scr@compatibility{\@nameuse{scr@v@#1}}%
⟨typearea⟩     \edef\scr@ta@compatibility{\@nameuse{scr@v@#1}}%
  }%
}
```

Eine zusätzliche Bedingung gibt es noch: Die Kompatibilität kann nur beim Laden gesetzt werden. Danach geht es nicht mehr:

```
⟨class⟩\AtEndOfClass{%
⟨package⟩\AtEndOfPackage{%
  \KOMA@key{version}[]{%
⟨class⟩     \ClassError{\KOMAClassName}{%
⟨package⟩     \PackageError{%
⟨extend⟩       scrextend%
⟨typearea⟩       typearea%
⟨package⟩     }{%
      Option ‘version’ too late%
    }{%
      Option ‘version’ may be set only while loading the
⟨class⟩       class.\MessageBreak
⟨package⟩       package.\MessageBreak
      But you’ve tried to set it up later.%
    }%
  }%
}
```

\scr@v@first
\scr@v@2.9
\scr@v@2.9t
\scr@v@2.95
\scr@v@2.95a
\scr@v@2.95b
\scr@v@2.96
\scr@v@2.96a
\scr@v@2.97
\scr@v@2.97a
\scr@v@2.97b
\scr@v@2.97c
\scr@v@2.97d
\scr@v@2.97e
\scr@v@2.98
\scr@v@2.98a
\scr@v@2.98b
\scr@v@2.98c
\scr@v@3.00
\scr@v@3.01
\scr@v@3.01a
\scr@v@3.01b
\scr@v@3.01c
\scr@v@3.02
\scr@v@3.02b
\scr@v@3.02c
\scr@v@3.03
\scr@v@3.03a
\scr@v@3.03b
\scr@v@3.04
\scr@v@3.05
\scr@v@3.04a

Nun die unterschiedlichen möglichen Werte (`\scr@v@last` ist jeweils die höchste vorhandene Nummer):

```
\@namedef{scr@v@first}{0}
\@namedef{scr@v@2.9}{0}
\@namedef{scr@v@2.9t}{0}
\@namedef{scr@v@2.9u}{1}
\@namedef{scr@v@2.95}{2}
\@namedef{scr@v@2.95a}{2}
\@namedef{scr@v@2.95b}{2}
\@namedef{scr@v@2.96}{2}
\@namedef{scr@v@2.96a}{3}
\@namedef{scr@v@2.97}{3}
\@namedef{scr@v@2.97a}{3}
\@namedef{scr@v@2.97b}{3}
\@namedef{scr@v@2.97c}{4}
\@namedef{scr@v@2.97d}{5}
\@namedef{scr@v@2.97e}{6}
\@namedef{scr@v@2.98}{6}
\@namedef{scr@v@2.98a}{6}
\@namedef{scr@v@2.98b}{6}
\@namedef{scr@v@2.98c}{7}
\@namedef{scr@v@3.00}{8}
```

```
\@namedef{scr@v@3.01}{8}
\@namedef{scr@v@3.01a}{8}
\@namedef{scr@v@3.01b}{9}
\@namedef{scr@v@3.01c}{9}
\@namedef{scr@v@3.02}{9}
\@namedef{scr@v@3.02b}{9}
\@namedef{scr@v@3.02c}{10}
\@namedef{scr@v@3.03}{10}
\@namedef{scr@v@3.03a}{10}
\@namedef{scr@v@3.03b}{10}
\@namedef{scr@v@3.04}{10}
\@namedef{scr@v@3.04a}{10}
\@namedef{scr@v@3.05}{10}
\@namedef{scr@v@3.05a}{10}
\@namedef{scr@v@3.06}{10}
\@namedef{scr@v@3.07}{10}
\@namedef{scr@v@3.08}{10}
\@namedef{scr@v@3.08a}{10}
\@namedef{scr@v@last}{10}
```

19.2 Kompatibilität mit früheren Versionen von scrlttr2

In früheren Versionen von scrlttr2 gab es weitere Befehle, die eventuell von `lco`-Dateien oder Paketen verwendet werden. Gemeldete Inkompatibitlitäten sind nach Möglichkeit zu lösen.

`\@setif` Dies war ein Makro, mit dem man einen Schalter über die symbolischen Werte `true`, `false`, `on` und `off` setzen kann. Das erste, optionale Argument war der Name des Schalters ohne Präfix „`if`“. Das zweite Argument war der Name der Option und das dritte der gewünschte Wert. War das optionale Argument nicht gesetzt oder leer, so wurde der Optionenname mit einem vorangestellten „`@`“ als Name des Schalters verwendet. Dieses Makro wird nun mit Hilfe von `\KOMA@set@ifkey` nachgebildet. Dadurch ist es nicht absolut fehlerkompatibel, das es nun mehr Werte versteht als vorher.

```
\expandafter\ifnum \@nameuse{scr@v@2.95}>\scr@compatibility\relax
  \newcommand*{\@setif}[2][]{%
    \begingroup
      \edef\@tempa{#1}\ifx\@tempa\@empty
        \def\@tempa{\KOMA@set@ifkey{#2}{@#2}}%
      \else
        \def\@tempa{\KOMA@set@ifkey{#2}{#1}}%
      \fi
    \expandafter\endgroup\@tempa
  }%
\fi
```

Datei t scrknpap.dtx

20 Briefbogen und Briefpapier

Bei Briefen kommt dem Briefbogen, also dem ersten Blatt des Briefes eine besondere Bedeutung zu. Dieser ist bei `scrlttr2` durch diverse Optionen, Variablen, Pseudo-Längen und Anweisungen bestimmt. Die weiteren Briefseiten werden hier verinfacht als das Briefpapier bezeichnet. Natürlich ist genau genommen auch der Briefbogen Briefpapier, aber eben ein besonderes.

20.1 Option für den Briefbogen

Gerade der Briefbogen ist durch eine ganze Anzahl von Optionen konfigurierbar. Daher wird hier weiter thematisch untergliedert.

20.1.1 Optionen für den Briefkopf

Die Gestaltung des Briefkopfes war bei der alten `scrlettr` Klasse eng begrenzt. Deshalb hatte ich für mich selbst das `briefkopf` Paket geschrieben und ohne Support veröffentlicht. Die Funktionalität ist nun direkt verfügbar. Dazu werden verschiedene Optionen benötigt.

`firsthead` Der Briefkopf kann auch einfach abgeschaltet werden.

`\if@firsthead` Gespeichert wird die Information in einem einfachen Schalter. Voreingestellt ist die Verwendung des Briefkopfes.

```
\KOMA@ifkey{firsthead}{@firsthead}\@firstheadtrue
```

`fromalign` Diese Option legt fest, ob der Absender linksbündig, mittig oder rechtsbündig gesetzt werden soll.

`\@alignfrom` Die Einstellung wird in einem nummerisch auszuwertenden Makro gespeichert. Dabei gelten folgende Werte:

Wert	Ausrichtung
0	linksbündig
1	zentriert
2	rechtsbündig
3	linksbündig in der Absenderergänzung ohne Kopf
4	rechtbündig in der Absenderergänzung ohne Kopf
10000	nicht antasten

Voreingestellt ist „nicht antasten“.

```
\newcommand*{\@alignfrom}{0}
\KOMA@key{fromalign}{%
  \KOMA@set@ncmdkey{fromalign}{@alignfrom}{%
    {left}{0},%
    {center}{1},{centered}{1},{middle}{1},%
    {right}{2},%
    {locationleft}{3},{leftlocation}{3},%
    {locationright}{4},{rightlocation}{4},{location}{4},%
    {no}{10000},{false}{10000},{off}{10000}%
  }{#1}%
  \ifcase\@alignfrom\or\or\or\@firstheadfalse\or\@firstheadfalse\fi
}
```

`fromrule` Es kann gewählt werden, ob im Absender nach dem Namen oder nach dem gesamten Absender eine Linie gesetzt werden soll.

`\@rulefrom` Die Einstellung wird in einem nummerisch auszuwertenden Makro gespeichert. Dabei gelten folgende Werte:

Wert	Absenderlinie
0	keine
1	unter dem Name
2	unter der Adresse

Voreingestellt ist keine Linie.

```
\newcommand*{\@rulefrom}{0}
\KOMA@ncmdkey{fromrule}[below]{@rulefrom}{%
  {false}{0},{no}{0},{off}{0},{aftername}{1},%
  {below}{2},{afteraddress}{2},{true}{2},{yes}{2},{on}{2}%
}
```

`fromphone` Ob die Telefonnummer im Absender gesetzt werden soll oder nicht, kann hier angegeben werden.

`\if@phone` `\@phonetrue` `\@phonefalse` Gespeichert wird die Information in einem einfachen Schalter. Voreingestellt ist keine Telefonnummer im Absender.

```
\KOMA@ifkey{fromphone}{@phone}
```

`fromfax` Ob die Faxnummer im Absender gesetzt werden soll oder nicht, kann hier angegeben werden.

`\if@fax` `\@faxtrue` `\@faxfalse` Gespeichert wird die Information in einem einfachen Schalter. Voreingestellt ist keine Faxnummer im Absender.

```
\KOMA@ifkey{fromfax}{@fax}
```

`fromemail` Ob die E-Mail-Adresse im Absender gesetzt werden soll oder nicht, kann hier angegeben werden.

\if@email
\@emailtrue
\@emailfalse

Gespeichert wird die Information in einem einfachen Schalter. Voreingestellt ist keine E-Mail-Adresse im Absender.

```
\KOMA@ifkey{fromemail}{@email}
```

fromurl

Ob die URL auf eine Homepage im Absender gesetzt werden soll oder nicht, kann hier angegeben werden.

\if@www
\@wwwtrue
\@wwwfalse

Gespeichert wird die Information in einem einfachen Schalter. Voreingestellt ist keine URL im Absender.

```
\KOMA@ifkey{fromurl}{@www}
```

fromlogo

Ob ein Logo in den Briefkopf gesetzt werden soll oder nicht, kann hier angegeben werden.

\if@logo
\@logotrue
\@logofalse

Gespeichert wird die Information in einem einfachen Schalter. Voreingestellt ist kein Logo im Briefkopf.

```
\KOMA@ifkey{fromlogo}{@logo}
```

20.1.2 Optionen für Anschrift und Absenderergänzung

addrfield

Das Adressfeld für Fensterumschläge kann mit `addrfield` ein- und ausgeschaltet werden. Darüber hinaus kann die Art des Adressfeldes gewählt werden.

\if@afield
\@afieldtrue
\@afieldfalse
\@afieldmode

Die Speicherung erfolgt wiederum in einem Makro.

```
\newcommand*{\if@afield}{%
  \ClassWarning{\KOMAClassName}{%
    Usage of \string\if@afield\space deprecated.\MessageBreak
    Internal macro \string\if@afield\space was removed at\MessageBreak
    KOMA-Script version 3.03.  I'll replace the usage of\MessageBreak
    \string\if@afield\space by \string\ifnum
    \string\@afieldmode=\string\@ne.\MessageBreak
    This may result in errors.  A better solution would be\MessageBreak
    to change the code}%
  \ifnum \@afieldmode=\@ne}
\newcommand*{\@afieldtrue}{%
  \ClassWarning{\KOMAClassName}{%
    Usage of \string\if@afield\space deprecated.\MessageBreak
    Internal macro \string\if@afield\space was removed at\MessageBreak
    KOMA-Script version 3.03.  I'll replace the usage of\MessageBreak
    \string\@afieldtrue\space by
    \string\KOMAoptions{addrfield=true}.\MessageBreak
    This may result in erros.  A better solution would be\MessageBreak
    to change the code}%
  \KOMAoptions{addrfield=true}}
\newcommand*{\@afieldfalse}{%
  \ClassWarning{\KOMAClassName}{%
    Usage of \string\if@afield\space deprecated.\MessageBreak
```

```
    Internal macro \string\if@afield\space was removed at\MessageBreak
    KOMA-Script version 3.03.  I'll replace the usage of\MessageBreak
    \string\@afieldfalse\space by
    \string\KOMAoptions{addrfield=false}.\MessageBreak
    This may result in erros.  A better solution would be\MessageBreak
    to change the code}%
  \KOMAoptions{addrfield=false}}
\newcommand*{\@afieldmode}{\@ne}
\KOMA@ncmdkey{addrfield}[true]{@afieldmode}{%
  {false}{0},{off}{0},{no}{0},%
  {true}{1},{on}{1},{yes}{1},%
  {PP}{2},{pp}{2},%
  {PPexplicite}{2},{PPExplicite}{2},%
  {ppexplicite}{2},{ppExplicite}{2},%
  {backgroundimage}{3},%
  {PPbackgroundimage}{3},{PPBackgroundImage}{3},{PPBackGroundImage}{3},%
  {ppbackgroundimage}{3},{ppBackgroundImage}{3},{ppBackGroundImage}{3},%
  {image}{4},{Image}{4},%
  {PPimage}{4},{PPImage}{4},%
  {ppimage}{4},{ppImage}{4}%
}
```

`priority` Im internationalen Briefverkehr sind zwei Prioritätseinstellungen bekannt. Dafür gibt es dann so etwas wie eine Versandart nicht mehr.

`\ltr@priority` Abgelegt wird die Art der Priorität in einem Macro.

```
\newcommand*{\ltr@priority}{0}%
\KOMA@ncmdkey{priority}[A]{ltr@priority}{%
  {no}{0},{off}{0},{false}{0},{manual}{1},%
  {B}{1},{b}{1},%
  {economy}{1},{Economy}{1},{ECONOMY}{1},%
  {B-ECONOMY}{1},{B-Economy}{1},{b-economy}{1},%
  {A}{2},{a}{2},%
  {priority}{2},{Priority}{2},{PRIORITY}{2},%
  {A-PRIORITY}{2},{A-Priority}{2},{a-priority}{2}%
}%
```

`backaddress` Im Adressfeld kann die Rücksendeadresse gesetzt werden. Dies geschieht nun nicht mehr allein durch Definition einer Rücksendeadresse, sondern auch durch eine Option.

`\if@baddr` `\@baddrtrue` `\@baddrfalse` `\backaddr@format` Dieses Makro hat genau ein Argument und bestimmt die Form der Rücksendeadresse. Voreingestellt ist Unterstreichung. Dies entspricht der vor Version 2.96 festen Voreinstellung.

```
\newcommand{\backaddr@format}[1]{}
\let\backaddr@format\underline
```

Die Einstellung true/false der Option wird in einem Schalter gespeichert.

```
\newif\if@baddr\@baddrtrue
```

```
\KOMA@key{backaddress}[true]{%
  \begingroup
    \KOMA@set@ncmdkey{backaddress}{@tempa}{%
      {false}{0},{off}{0},{no}{0},%
      {true}{1},{on}{1},{yes}{1},%
      {underlined}{2},%
      {plain}{3}%
    }{#1}%
    \ifcase\@tempa
      \endgroup
      \@baddrfalse
    \or
      \endgroup
      \@baddrtrue
    \or
      \endgroup
      \@baddrtrue
      \let\backaddr@format\underline
    \or
      \endgroup
      \@baddrtrue
      \let\backaddr@format\@firstofone
    \else
      \endgroup
    \fi
}
```

`locfield` Mit Hilfe der Optione `locfield` kann der Platz neben dem Adressfeld groß oder klein gewählt werden.

`\if@bigloc` `\@bigloctrue` `\@biglocfalse` Die Einstellung wird in einem Schalter gespeichert.

```
\newif\if@bigloc
```

```
\KOMA@key{locfield}[wide]{%
  \begingroup
    \KOMA@set@ncmdkey{locfield}{@tempa}{{wide}{0},{narrow}{1}}{#1}%
    \ifcase\@tempa
      \endgroup
      \@bigloctrue
    \else
      \endgroup
      \@biglocfalse
    \fi
}
```

`wlocfield` `slocfield`

```
\KOMA@DeclareDeprecatedOption{wlocfield}{locfield=wide}
\KOMA@DeclareDeprecatedOption{slocfield}{locfield=narrow}
```

20.1.3 Optionen für Faltmarken

`foldmarks` Die Faltmarken können mit `foldmarks` aktiviert und deaktiviert werden.

`\if@fold` `\@foldtrue` `\@foldfalse` Ob Faltmarken gesetzt werden sollen oder nicht, wird in einem Schalter gespeichert.

```
\newif\if@fold\@foldtrue
```

`\@usefolds` Welche Faltmarken gesetzt werden sollen, wird in einem Makro gespeichert.

```
\newcommand*{\@usefolds}{TMBPL}
\KOMA@key{foldmarks}[true]{%
  \begingroup
    \def\FamilyUnknownKeyValue##1##2##3##4{%
      \def\sc@nf@ldch@rs####1{%
        \ifx ####1\@nil\else
          \if ####1T\g@addto@macro\@usefolds{####1}\else
            \if ####1M\g@addto@macro\@usefolds{####1}\else
              \if ####1B\g@addto@macro\@usefolds{####1}\else
                \if ####1P\g@addto@macro\@usefolds{####1}\else
                  \if ####1L\g@addto@macro\@usefolds{####1}\else
                    \if ####1V\g@addto@macro\@usefolds{L}\else
                      \if ####1H\g@addto@macro\@usefolds{TMBP}\else
                        \if ####1t\scr@removechar\@usefolds{T}\else
                          \if ####1m\scr@removechar\@usefolds{M}\else
                            \if ####1b\scr@removechar\@usefolds{B}\else
                              \if ####1p\scr@removechar\@usefolds{P}\else
                                \if ####1l\scr@removechar\@usefolds{L}\else
                                  \if ####1v%
                                    \scr@removechar\@usefolds{L}%
                                  \else
                                    \if ####1h%
                                      \scr@removechar\@usefolds{T}%
                                      \scr@removechar\@usefolds{M}%
                                      \scr@removechar\@usefolds{B}%
                                    \else
                                      \ClassError{\KOMAClassName}{%
                                        unknown fold mark switch ‘####1’}{%
                                        You should use option ‘foldmarks’
                                        either with one of\MessageBreak
                                        \@spaces ‘true’, ‘on’, or ‘yes’ to
                                        switch fold marks on,\MessageBreak
                                        \@spaces ‘false’, ‘off’, or ‘no’ to
                                        switch fold marks off,\MessageBreak
                                        or a combination of\MessageBreak
                                        \@spaces ‘T’, ‘M’, ‘B’, ’P’, ’L’, ‘V’,
                                        ‘H’ to select dedicated fold
                                        marks\MessageBreak
                                        or\MessageBreak
```

```
                                        \@spaces ‘t’, ‘m’, ’b’, ’p’, ’l’, ’v’,
                                        ’h’ to deselect dedicated fold
                                        marks.\MessageBreak
                                        See the KOMA-Script manual for more
                                        information.
                                      }%
                                    \fi
                                  \fi
                                \fi
                              \fi
                            \fi
                          \fi
                        \fi
                      \fi
                    \fi
                  \fi
                \fi
              \fi
            \fi
            \fi
            \expandafter\sc@nf@ldch@rs
          \fi
        }%
        \expandafter\sc@nf@ldch@rs#1\@nil
      }%
      \KOMA@set@ifkey\@empty{@fold}{#1}%
      \if@fold\aftergroup\@foldtrue\else\aftergroup\@foldfalse\fi
    \endgroup
}
```

\scr@removechar

```
\newcommand*{\scr@removechar}[2]{%
  \def\in@@##1#2##2##3\in@@{%
    \ifx\in@##2\else
      \def\in@@####1#2\in@{\gdef#1{##1####1}}%
      \in@@##2##3%
    \fi}%
  \expandafter\in@@#1#2\in@\in@@
}
```

20.1.4 Optionen für die Geschäftszeile

`refline` Kopf und Fuß der ersten Seite sind normalweise so breit, dass sie zur Adresse des Empfängers passen. Für die Geschäftszeile gibt es zwei Alternativen, eine breite und eine, deren Breite der Textbreite entspricht. Voreingestellt ist die Textbreite.

`\if@refwide` `\@refwidetrue` `\@refwidefalse` Gespeichert wird die Information in einem Schalter.

```
\newif\if@refwide
```

```
\KOMA@key{refline}[wide]{%
  \begingroup
    \KOMA@set@ncmdkey{refline}{@tempa}{{wide}{0},{narrow}{1}}{#1}%
    \ifcase\@tempa
      \endgroup
      \@refwidetrue
    \else
      \endgroup
      \@refwidefalse
    \fi
}
```

20.1.5 Optionen für den Betreff

`subject` Ein Betreff-Titel kann mit `subject` ein- und abgeschaltet werden. In England und Frankreich scheint es teilweise üblich zu sein, den Betreff erst nach der Anrede und dafür zentriert zu setzen[3]. Deshalb kann alternativ zum Einschalten des Betreff-Titels hier auch dies aktiviert werden.

`\subject@format` Das Format des Betreffs kann geändert werden.

```
\newcommand{\subject@format}[1]{#1}
```

`\raggedsubject`

```
\newcommand*{\raggedsubject}{}
```

`\if@subj` `\@subjtrue` `\@subjfalse` `\if@subjafter` `\@subjaftertrue` `\@subjafterfalse` Die Speicherung erfolgt in zwei Schaltern.

```
\newif\if@subj
\newif\if@subjafter
\KOMA@key{subject}{%
  \begingroup
    \KOMA@set@ncmdkey{subject}{@tempa}{%
      {titled}{0},%
      {untitled}{1},%
      {afteropening}{2},%
      {beforeopening}{3},%
      {underlined}{4},%
      {left}{5},%
      {right}{6},%
      {centered}{7}%
    }{#1}%
    \ifcase\@tempa
      \aftergroup\@subjtrue
    \or
      \aftergroup\@subjfalse
    \or
      \aftergroup\@subjaftertrue
    \or
```

[3] Ich kann mir nur mühsahm eine Bemerkung dazu verkneifen.

```
      \aftergroup\@subjafterfalse
    \or
      \aftergroup\let\aftergroup\subject@format\aftergroup\underline
    \or
      \aftergroup\let\aftergroup\raggedsubject\aftergroup\raggedright
    \or
      \aftergroup\let\aftergroup\raggedsubject\aftergroup\raggedleft
    \or
      \aftergroup\let\aftergroup\raggedsubject\aftergroup\centering
    \fi
  \endgroup
}
```

20.1.6 Optionen für den Satzspiegel

`enlargefirstpage` Normalerweise wird der Text des Briefboges entsprechend dem normalen Satzspiegel gesetzt. Ist allerdings der Fuß der ersten Seite zu hoch, dann wird der Satzspiegel vertikal entsprechend verkürzt. Mit der Option `firstpageenlarge` kann erzwungen werden, dass im umgekehrten Fall, der Satzspiegel vertikal verlängert wird.

`\if@enlargefp` `\@enlargefptrue` `\@enlargefpfalse` Gespeichert wird die Information über die Optionen in einem Schalter.

```
\KOMA@ifkey{enlargefirstpage}{@enlargefp}
```

20.1.7 Optionen für den Briefbogenfuß

`firstfoot` Der Fuß des Briefbogens kann auch einfach abgeschaltet werden.

`\if@firstfoot` Gespeichert wird die Information in einem einfachen Schalter. Voreingestellt ist die Verwendung des Fußes.

```
\KOMA@ifkey{firstfoot}{@firstfoot}\@firstfoottrue
```

20.2 Optionen für den Kopf des weiteren Briefpapiers

Diese Optionen haben zwar nichts mit dem eigentlichen Briefbogen zu tun. Das Briefpapier schließt sich aber an, weshalb die Optionen hier zu finden sind.

`pagenumber` Bei Briefen kann gewählt werden, ob auf Folgeseiten die Seitenzahl im Kopf oder im Fuß stehen soll. Dabei kann zwischen Seitenzahl rechts, links oder zentriert gewählt werden. Voreingestellt ist der Fuß und mittig.

`\@pageat` In dieser nummerisch auswertbaren Variablen wird gespeichert, wo die Seitenzahl stehen soll. Dabei gilt:

Wert	Bedeutung für die Seitenzahl
-1	keine
0	im Kopf links
1	im Kopf mittig
2	im Kopf rechts
3	im Fuß links
4	im Fuß mittig
5	im Fuß rechts

```
\newcommand*{\@pageat}{4}
\KOMA@key{pagenumber}{%
  \begingroup
    \KOMA@set@ncmdkey{pagenumber}{@tempa}{%
      {no}{-1},{off}{-1},{false}{-1},%
      {topleft}{0},{headleft}{0},%
      {topmiddle}{1},{topcenter}{1},{topcentered}{1},%
      {headmiddle}{1},{headcenter}{1},{headcentered}{1},%
      {topright}{2},{headright}{2},%
      {botleft}{3},{footleft}{3},%
      {botmiddle}{4},{botcenter}{4},{botcentered}{4},%
      {footmiddle}{4},{footcenter}{4},{footcentered}{4},%
      {botright}{5},{footright}{5},%
      {top}{10},{head}{10},{bot}{11},{foot}{11},%
      {left}{20},{middle}{21},{center}{21},{centered}{21},{right}{22}%
    }{#1}%
    \ifnum\@tempa<10
      \edef\@tempa{\endgroup\edef\noexpand\@pageat{\@tempa}}%
    \else
      \ifnum\@tempa<20
        \@tempcnta\@pageat\relax
        \ifnum\@tempcnta>2
          \advance\@tempcnta by -3
        \fi
        \ifnum\@tempa=11
          \advance\@tempcnta by 3
        \fi
        \edef\@tempa{\endgroup\edef\noexpand\@pageat{\the\@tempcnta}}%
      \else
        \ifnum\@pageat>2
          \edef\@pageat{3}%
        \else
          \edef\@pageat{0}%
        \fi
        \@tempcnta\@tempa
        \advance\@tempcnta by -20
        \advance\@tempcnta by\@pageat
        \edef\@tempa{\endgroup\edef\noexpand\@pageat{\the\@tempcnta}}%
      \fi
    \fi
```

```
  \@tempa
}
```

20.3 Anweisungen für den Briefbogen

Auch bei den Anweisungen zunächst alles, was mit dem Briefbogen zu tun hat.

20.3.1 Der Kopf des Briefbogens

\firsthead
\@firsthead
\@gen@firsthead
firsthead

Bei scrlttr2 gibt es die üblichen Seitenstile, allerdings ist deren Verhalten etwas anders als gewohnt. Dies ist in den Besonderheiten für Briefen begründet. Zunächst die Voreinstellung für Briefbögen.

```
\newcommand*{\@firsthead}{%
  \ClassWarning{scrlttr2}{%
    Deprecated usage of \string\@firsthead.\MessageBreak
    \string\usekomavar{firsthead} should be used\MessageBreak
    instead of \string\@firsthead
  }%
  \usekomavar{firsthead}%
}
\newkomavar{firsthead}
\setkomavar{firsthead}{%
  \parbox[b]{\useplength{firstheadwidth}}%
  {\centering\usekomafont{fromaddress}{%
      \ifkomavarempty{fromname}{}{%
        {\usekomafont{fromname}%
          {\ignorespaces\usekomavar{fromname}}}\\[-8pt]
        {\usekomafont{fromrule}{\rule{\useplength{firstheadwidth}}{.4pt}}}\\
      }%
      \ifkomavarempty{fromaddress}{}{%
        \ignorespaces\usekomavar{fromaddress}%
      }%
    }%
  }%
}
```

\scr@firsthead@postsetvar

Über dieses Makro wird jeweils nach dem Setzen der Variable die Ausrichtung auf manuell umgeschaltet. Das darf deshalb auch erst nach dem erstmaligen Setzen definiert werden!

```
\newcommand*{\scr@firsthead@postsetvar}{%
  \renewcommand*{\@alignfrom}{10000}%
}
```

```
\newcommand*{\@gen@firsthead}{%
```

Damit wird ggf. die Kopfzeile der ersten Seite umdefiniert, nämlich dann, wenn dies per Option aktiviert wurde.

```
  \ifcase\@alignfrom
```

Wir befinden uns nun in dem Bereich für `\@alignfrom`=0. Entsprechend unserer Festlegung wird daher als Erstes der linksbündige Absender definiert. Wir verwenden dafür eine *parbox*, in der der Text linksbündig angeordnet wird. Diese Box soll über die gesamte Breite gehen dürfen:

```
    \setkomavar{firsthead}{%
      \parbox[b]{\textwidth}{\raggedright
```

Zuerst wird der Name gesetzt. Dabei soll eine eigene Schriftart für den Namen verwendet werden.

```
        {\usekomafont{fromname}\strut\ignorespaces
          \usekomavar{fromname}%
```

Hat `\@rulefrom` den nummerischen Wert 1, so soll direkt auf den Absender eine Linie folgen. Die Länge dieser Linie kann vom Anwender bestimmt oder automatisch gewählt werden. Es werden dabei die Pseudo-Längen `fromrulewidth` und `fromrulethickness` verwendet. Ist `fromrulewidth` negativ, so wird sie automatisch bestimmt. Bei der automatischen Längenbestimmung wird dafür gesorgt, dass die Linie nicht bis in das Logo ragt, das eventuell rechtsbündig neben dem Absender steht:

```
          {%
            \ifnum\@rulefrom=1\\[-.5\baselineskip]%
              \usekomafont{fromrule}{%
                \setlength{\@tempdima}{\useplength{fromrulewidth}}%
                \ifdim\@tempdima=\z@
                  \setlength{\@tempdima}{\textwidth}%
                  \if@logo
                    \settowidth{\@tempdimb}{\usekomavar{fromlogo}}%
                    \addtolength{\@tempdima}{-\@tempdimb}%
                  \fi
                \fi
                \expandafter\rule\expandafter{\the\@tempdima}{%
                  \useplength{fromrulethickness}}}%
            \fi
          }%
          \\
        }%
```

Als Nächstes wird die Absenderadresse einschließlich aller optionalen Felder gesetzt. Dabei werden optionale Felder nur gesetzt, wenn der entsprechende Schalter *true* ist. Die Adresse wird ebenfalls innerhalb einer Gruppe in einer eigenen Schriftart gesetzt.

```
        {\usekomafont{fromaddress}\strut\ignorespaces
          \usekomavar{fromaddress}%
          \if@phone\\\usekomavar*{fromphone}\usekomavar{fromphone}\fi%
          \if@fax\\\usekomavar*{fromfax}\usekomavar{fromfax}\fi%
          \if@email\\\usekomavar*{fromemail}\usekomavar{fromemail}\fi%
          \if@www\\\usekomavar*{fromurl}\usekomavar{fromurl}\fi%
```

Wie weiter oben erwähnt, kann auch an dieser Stelle eine Linie gesetzt werden. Wir gehen davon aus, dass die Linie automatisch nicht in das Logo

ragt, da es in diesem Fall über der Linie zu stehen hat. Die entsprechende Berechnung kann also entfallen.

```
          \ifnum\@rulefrom=2\\[-.5\baselineskip]%
            \usekomafont{fromrule}{%
              \setlength{\@tempdima}{\useplength{fromrulewidth}}%
              \ifdim\@tempdima=\z@
                \setlength{\@tempdima}{\textwidth}%
              \fi
              \expandafter\rule\expandafter{\the\@tempdima}{%
                \useplength{fromrulethickness}}}%
          \fi
          \\
        }%
      }%
```

Die parbox ist damit ebenfalls beendet. Es muss nun ggf. noch das Logo gesetz werden. Es wird in diesem Fall von der aktuellen Position am rechten Rand nach links ausgegeben.

```
      \if@logo
        \llap{\usekomavar{fromlogo}}%
      \fi
    }%
    \renewcommand*{\@alignfrom}{0}%
```

Damit ist die Definition des linksbündigen Absenders abgeschlossen. Es geht weiter mit dem zentrierten Absender. Dieser hat als Besonderheit, dass das Logo nicht in den Briefkopf, sondern in den Bereich rechts neben dem Empfänger gesetzt wird. Dort ist bereits das location-Feld, daher wird dieser Fall auch dort behandelt.

```
  \or
    \setkomavar{firsthead}{%
      \parbox[b]{\textwidth}{\centering
        {\usekomafont{fromname}\strut\ignorespaces
          \usekomavar{fromname}%
          {%
            \ifnum\@rulefrom=1\\[-.5\baselineskip]%
              \usekomafont{fromrule}{%
                \setlength{\@tempdima}{\useplength{fromrulewidth}}%
                \ifdim\@tempdima=\z@
                  \setlength{\@tempdima}{\textwidth}%
                \fi
                \expandafter\rule\expandafter{\the\@tempdima}{%
                  \useplength{fromrulethickness}}}%
            \fi
          }%
          \\
        }%
        {\usekomafont{fromaddress}\strut\ignorespaces
          \usekomavar{fromaddress}%
```

```
          \if@phone\\\usekomavar*{fromphone}\usekomavar{fromphone}\fi%
          \if@fax\\\usekomavar*{fromfax}\usekomavar{fromfax}\fi%
          \if@email\\\usekomavar*{fromemail}\usekomavar{fromemail}\fi%
          \if@www\\\usekomavar*{fromurl}\usekomavar{fromurl}\fi%
          \ifnum\@rulefrom=2\\[-.5\baselineskip]%
            \usekomafont{fromrule}{%
              \setlength{\@tempdima}{\useplength{fromrulewidth}}%
              \ifdim\@tempdima=\z@
                \setlength{\@tempdima}{\textwidth}%
              \fi
              \expandafter\rule\expandafter{\the\@tempdima}{%
                \useplength{fromrulethickness}}}%
          \fi
          \\
        }%
      }%
    }%
    \renewcommand*{\@alignfrom}{1}%
```

Damit ist die Definition des zentrierten Absenders abgeschlossen. Es geht weiter mit dem rechtsbündigen Absender. Das ganze wird einfach beim linksbündigen Absender abgeschaut.

```
  \or
    \setkomavar{firsthead}{%
      \if@logo
        \rlap{\usekomavar{fromlogo}}%
      \fi
      \parbox[b]{\textwidth}{\raggedleft
        {\usekomafont{fromname}\strut\ignorespaces
          \usekomavar{fromname}%
          {%
            \ifnum\@rulefrom=1\\[-.5\baselineskip]%
              \usekomafont{fromrule}{%
                \setlength{\@tempdima}{\useplength{fromrulewidth}}%
                \ifdim\@tempdima=\z@
                  \setlength{\@tempdima}{\textwidth}%
                  \if@logo
                    \settowidth{\@tempdimb}{\usekomavar{fromlogo}}%
                    \addtolength{\@tempdima}{-\@tempdimb}%
                  \fi
                \fi
                \expandafter\rule\expandafter{\the\@tempdima}{%
                  \useplength{fromrulethickness}}}%
            \fi
          }%
          \\
        }%
        {\usekomafont{fromaddress}\strut\ignorespaces
          \usekomavar{fromaddress}%
          \if@phone\\\usekomavar*{fromphone}\usekomavar{fromphone}\fi%
```

```
          \if@fax\\\usekomavar*{fromfax}\usekomavar{fromfax}\fi%
          \if@email\\\usekomavar*{fromemail}\usekomavar{fromemail}\fi%
          \if@www\\\usekomavar*{fromurl}\usekomavar{fromurl}\fi%
          \ifnum\@rulefrom=2\\[-.5\baselineskip]%
            \usekomafont{fromrule}{%
              \setlength{\@tempdima}{\useplength{fromrulewidth}}%
              \ifdim\@tempdima=\z@
                \setlength{\@tempdima}{\textwidth}%
              \fi
              \expandafter\rule\expandafter{\the\@tempdima}{%
                \useplength{fromrulethickness}}}%
          \fi
          \\
        }%
      }%
    }%
    \renewcommand*{\@alignfrom}{2}%
```

Damit ist die Umdefinierung beendet. In allen anderen Fällen wird der Kopf nicht verändert.

```
  \fi
}
```

Der Benutzer kann die Voreinstellung mit einer einfachen Anweisung ändern:

```
\newcommand{\firsthead}{%
  \ClassWarning{scrlttr2}{%
    Deprecated usage of \string\firsthead.\MessageBreak
    You should use \string\setkomavar{firsthead}\MessageBreak
    instead of \string\firsthead
  }
  \setkomavar{firsthead}%
}
```

20.3.2 Schrifteinstellungen für den Briefbogen

`foldmark` Die Schrift (hier ist die Farbe der entscheidende Faktor) der Faltmarken.

```
\newkomafont{foldmark}{}
```

`fromname` Der Absendername im Briefkopf.

```
\newkomafont{fromname}{}
```

`fromaddress` Die Absenderadresse im Briefkopf.

```
\newkomafont{fromaddress}{}
```

`fromrule` Die Schrift (hier ist die Farbe der entscheidende Faktor) der Linie im Absender.

```
\newkomafont{fromrule}{}
```

`backaddress` Die Rücksendeadresse.

```
\newkomafont{backaddress}{\sffamily}
```

`specialmail` Die Versandart.

```
\newkomafont{specialmail}{}
```

`priority` Grundschrift für die Priorität des Briefs, falls diese A oder B ist. Es wird ohne Durchschuss gearbeitet.

```
\newkomafont{priority}{\fontsize{10pt}{10pt}\sffamily\bfseries}
```

`prioritykey` Schrift für den Buchstaben der Priorität des Briefs, falls diese A oder B ist. Wird nach `priority` angewandt. Eigentlich sollten hier 9 mm verwendet werden. Dafür wird aber eine frei skalierbare Schrift benötig. Mit 24,88 pt sind wir nur 0,25 mm kleiner. Das sollte kein Problem darstellen. Es wird auch hier ohne Durchschuss gearbeitet.

```
\newkomafont{prioritykey}{\fontsize{24.88pt}{24.88pt}\selectfont}
```

`PPlogo` Schrift für das Logo `P.P.`, das im Anschriftmodus `PP` gesetzt wird.

```
\newkomafont{PPlogo}{\sffamily\bfseries}
```

`PPdata` Schrift, die innerhalb des PP-Kopfes für die Daten verwendet wird. gesetzt wird.

```
\newkomafont{PPdata}{\sffamily}
```

`addressee` Die komplette Anschrift.

```
\newkomafont{addressee}{}
```

`toname` Der Name des Empfängers in der Anschrift.

```
\newkomafont{toname}{}
```

`toaddress` Die Adresse des Empfängers in der Anschrift.

```
\newkomafont{toaddress}{}
```

`refname` Die Feldnamen in der Geschäftszeile.

```
\newkomafont{refname}{\sffamily\scriptsize}
```

`refvalue` Die Feldwerte in der Geschäftszeile.

```
\ifnum\scr@compatibility=\@nameuse{scr@v@2.97}\relax
  \newkomafont{refvalue}{\sffamily\scriptsize}
\else
  \newkomafont{refvalue}{}
\fi
```

`title` Der Brieftitel.

```
\newkomafont{title}{\normalcolor\sffamily\bfseries}
```

`subject` Der Betreff.

```
\newkomafont{subject}{\normalcolor\bfseries}
```

20.3.3 Längen für den Briefbogen

`\ltr@len@foldmarkhpos` Abstand der horizontalen Faltmarken von der linken Papierkante.

```
\@newplength{foldmarkhpos}
```

`\ltr@len@foldmarkvpos` Abstand der vertikalen Faltmarken von der oberen Papierkante.

```
\@newplength{foldmarkvpos}
```

`\ltr@len@tfoldmarkvpos` `\ltr@len@mfoldmarkvpos` `\ltr@len@bfoldmarkvpos` Abstand der oberen, mittleren und unteren Faltmarken von der oberen Papierkante.

```
\@newplength{tfoldmarkvpos}
\@newplength{mfoldmarkvpos}
\@newplength{bfoldmarkvpos}
```

`\ltr@len@lfoldmarkhpos` Abstand linken Faltmarken von der linken Papierkante.

```
\@newplength{lfoldmarkhpos}
```

`\ltr@len@pfoldmarklength` `\ltr@len@hfoldmarklength` `\ltr@len@mfoldmarklength` `\ltr@len@bfoldmarklength` `\ltr@len@lfoldmarklength` Länge der Lochmarke und der oberen, mittleren und unteren horizontalen und der (linken) vertikalen Faltmarke.

```
\@newplength{pfoldmarklength}
\@newplength{tfoldmarklength}
\@newplength{mfoldmarklength}
\@newplength{bfoldmarklength}
\@newplength{lfoldmarklength}
```

`\ltr@len@foldmarkthickness` Dicke der Faltmarken.

```
\@newplength{foldmarkthickness}
\@setplength{foldmarkthickness}{.2pt}
```

`\ltr@len@toaddrvpos` Abstand des Adressfensters von der oberen Papierkante.

```
\@newplength{toaddrvpos}
```

`\ltr@len@backaddrheight` Höhe der Rücksendeadresse.

```
\@newplength{backaddrheight}
```

`\ltr@len@toaddrhpos` Abstand des Adressfeldes von der linken Papierkante bzw. wenn negativ von der rechten Papierkante.

```
\@newplength{toaddrhpos}
```

`\ltr@len@specialmailindent` Abstand der Versandtart vom linken Rand des Adressfensters.

```
\@newplength{specialmailindent}
```

`ltr@len@specialmailrightindent` Abstand der Versandtart vom rechten Rand des Adressfensters.

```
\@newplength{specialmailrightindent}
```

`\ltr@len@toaddrindent` Abstand der Adresse vom linken Rand des Adressfensters. 0 bedeutet im Anschriftmodus `PP` 8 mm, ansonsten wirklich 0 mm. Außerdem wird im Anschriftmodus `PP` die Pseudolänge auch für den Abstand vom rechten Rand des Adressfensters verwendet.

```
\@newplength{toaddrindent}
```

`\ltr@len@PPheadheight` Höhe des PP-Kopfes in den Anschriftmodi `PP`, `backgroundimage` und `image`. 0 bedeutet hier 20,74 pt.

```
\@newplength{PPheadheight}
```

`\ltr@len@PPheadwidth` Breite des PP-Kopfes im Anschriftmodus `PP`. 0 bedeutet hier 42 mm.

```
\@newplength{PPheadwidth}
```

`\ltr@len@PPdatamatrixvskip` Abstand zwischen PP-Kopf und DataMatrix im Anschriftmodus `PP`. 0 bedeutet hier 9 mm.

```
\@newplength{PPdatamatrixvskip}
```

`\ltr@len@toaddrwidth` Breite des Adressfensters

```
\@newplength{toaddrwidth}
```

`\ltr@len@toaddrheight` Höhe des Adressfensters mit Rücksendeadresse

```
\@newplength{toaddrheight}
```

`\ltr@len@locvpos` Abstand der Absenderergänzung von der oberen Papierkante.

```
\@newplength{locvpos}
```

`\ltr@len@locwidth` Breite des Feldes neben dem Adressfenster

```
\@newplength{locwidth}
```

`\ltr@len@lochpos` Abstand der Absenderergänzung von der linken Papierkante bzw. wenn negativ von der rechten Papierkante.

```
\@newplength{lochpos}
```

`\ltr@len@locheight` Höhe der Absenderergänzung.

```
\@newplength{locheight}
```

`\ltr@len@refvpos` Abstand der Geschäftszeile von der Papieroberkante.

```
\@newplength{refvpos}
```

`\ltr@len@refwidth` Breite der Geschäftszeile.

```
\@newplength{refwidth}
```

`\ltr@len@refhpos` Horizontale Position der Geschäftszeile.

```
\@newplength{refhpos}
```

`\ltr@len@refaftervskip` Abstand nach der Geschäftszeile.

```
\@newplength{refaftervskip}
```

\ltr@len@sigbeforevskip Raum für die Unterschrift.

```
\@newplength{sigbeforevskip}
```

\ltr@len@sigindent Horizontaler Einzug des Schlussgrußes und der Signatur.

```
\@newplength{sigindent}
```

\ltr@len@@firstfootvpos Abstand des Fußes der ersten Seite von der Papieroberkante.

```
\@newplength{firstfootvpos}
```

\ltr@len@firstfootwidth Breite des Fußes der ersten Seite.

```
\@newplength{firstfootwidth}
```

\ltr@len@firstfoothpos Abstand des Kopfes vom linken Rand der Seite.

```
\@newplength{firstfoothpos}
\@setplength{firstfoothpos}{\maxdimen}
```

\ltr@len@firstheadvspos Abstand des Kopfes der ersten Seite von der Papieroberkante.

```
\@newplength{firstheadvpos}
```

\ltr@len@firstheadwidth Breite des Kopfes der ersten Seite.

```
\@newplength{firstheadwidth}
```

\ltr@len@firstheadhpos Abstand des Kopfes vom linken Rand der Seite.

```
\@newplength{firstheadhpos}
\@setplength{firstheadhpos}{\maxdimen}
```

\ltr@len@fromrulewidth Die Länge einer eventuell gesetzten Linie im Absender. Negative Werte bedeuten, dass die Länge automtaisch bestimmt wird.

```
\@newplength{fromrulewidth}
\@setplength{fromrulewidth}{0pt}
```

\ltr@len@fromrulethickness Die Dicke einer eventuell gesetzte Linie im Absender.

```
\@newplength{fromrulethickness}
\@setplength{fromrulethickness}{.4pt}
```

20.3.4 Fuß des Briefbogens

\firstfoot
\@firstfoot
firstfoot

```
% Der Fuß des Briefbogens ist in der Voreinstellung leer.
\newkomavar{firstfoot}
\newcommand*{\@firstfoot}{%
  \ClassWarning{scrlttr2}{%
    Deprecated usage of \string\@firstfoot.\MessageBreak
    \string\usekomavar{firstfoot} should be used\MessageBreak
    instead of \string\@firstfoot
  }%
  \usekomavar{firstfoot}%
}
```

Dies kann jedoch über eine einfache Benutzeranweisung geändert werden:

```
\newcommand{\firstfoot}{%
  \ClassWarning{scrlttr2}{%
    Deprecated usage of \string\firstfoot.\MessageBreak
    You should use \string\setkomavar{firstfoot}\MessageBreak
    instead of \string\firstfoot
  }
  \setkomavar{firstfoot}%
}
```

20.3.5 Ausgabe des Briefbogens

Die Ausgabe selbst erfolgt derzeit über die `\opening`-Anweisung, die in einem späteren Abschnitt implementiert ist.

`\move@topt` Für die folgenden Aktionen wird ein Makro benötigt, mit dem man vertikal von der relativen Seitenposition zur relativen Papierposition gelangen kann.

```
\newcommand*{\move@topt}{%
  \null\vskip -1in\vskip -\topmargin%
  \vskip -\headheight\vskip -\headsep%
  \vskip -\topskip%
  \ignorespaces%
}
```

`\move@frompt` Natürlich wird auch ein Makro benötigt, um vertikal von einer relativen Papierposition wieder auf die relative Seitenposition zurück zu gelangen.

```
\newcommand*{\move@frompt}{%
  \vskip 1in\vskip \topmargin%
  \vskip \headheight\vskip \headsep%
  \vskip \topskip%
  \vskip -\baselineskip%
  \ignorespaces%
}
```

`\move@topl` Für die folgenden Aktionen wird außerdem ein Makro benötigt, mit dem man horizontal von der relativen Seitenposition zur relativen Papierposition gelangen kann.

```
\newcommand*{\move@topl}{%
  \null\hskip -1in%
  \ifodd\value{page}\hskip -\oddsidemargin\else\hskip -\evensidemargin\fi
  \ignorespaces%
}
```

`\move@frompl` Natürlich wird auch ein Makro benötigt, um horizontal von einer relativen Papierposition wieder auf die relative Seitenposition zurück zu gelangen.

```
\newcommand*{\move@frompl}{%
  \hskip 1in%
  \ifodd\value{page}\hskip \oddsidemargin\else\hskip \evensidemargin\fi%
```

```
  \ignorespaces%
}
```

\vb@t@z Ausserdem wird ein vertikales Äquivalent zu `\rlap` benötigt.

```
\newcommand{\vb@t@z}[1]{{%
    \setbox0\vbox to\z@{#1\vss}%
    \setparsizes{\z@}{\z@}{\z@ plus 1fil}\par@updaterelative
    \setlength{\baselineskip}{\z@}%
    \setlength{\@tempdima}{\ht0}\addtolength{\@tempdima}{\dp0}%
    \box0\vskip-\@tempdima%
  }%
}
```

\@foldmarks Für die Faltmarken müssen nicht nur Abstände definiert werden, es wird auch ein Befehl benötigt, um die Faltmarken zu setzen.

\@hfoldmark Dabei werden nur die horizontalen Faltmarken gesetzt, die tatsächlich existieren.

\@foldmark Das geht am einfachsten, wenn wir auch einen Befehl haben, mit dem man eine einzelne Faltmarke relativ zur aktuellen Position setzen kann. Dies geschieht mit Hilfe von `\rlap`, damit die horizontale Position unverändert bleibt. Als Parameter werden der vertikale Abstand zur aktuellen Position und die Länder der Marke übergeben.

```
\newcommand*{\@foldmark}[2]{%
  \vb@t@z{\setlength{\@tempdima}{#1}%
    \addtolengthplength[-.5]{\@tempdima}{foldmarkthickness}%
    \vskip \@tempdima
    \rlap{\move@topl\hskip\useplength{foldmarkhpos}%
      \usekomafont{foldmark}{%
        \rule{#2}{\useplength{foldmarkthickness}}}}}%
  \ignorespaces
}
\newcommand*{\@hfoldmark}[1]{%
  \begingroup
    \@expandtwoargs\in@{#1}{\@usefolds}%
    \ifin@
      \lowercase{%
        \edef\@tempa{%
          \noexpand\ifdim \noexpand\useplength{#1foldmarkvpos}=\noexpand\z@
          \noexpand\else
            \noexpand\@foldmark{%
              \noexpand\useplength{#1foldmarkvpos}%
            }{%
              \noexpand\useplength{#1foldmarklength}%
            }%
          \noexpand\fi
        }%
      }%
```

```
      \@tempa
    \fi
  \endgroup
}
```

\@vfoldmark Für die vertikale(n) Faltmarke(n) gilt prinzipiell das gleiche. Dabei ist lediglich zu beachten, dass die Linie hier komplett oberhalb der Grundlinie gesetzt wird, also die vertikale Position um die Länge nach unten zu verschieben ist.

```
\newcommand*{\@vfoldmark}[1]{%
  \begingroup
    \@expandtwoargs\in@{#1}{\@usefolds}%
    \ifin@
      \ifdim \useplength{lfoldmarkhpos}=\z@\else
        \vb@t@z{%
          \ifdim \useplength{foldmarkvpos}<\z@
            \nobreak\vskip \paperheight\nobreak
          \fi
          \vskip \useplength{foldmarkvpos}%
          \vskip \useplength{foldmarkthickness}%
          \rlap{\move@topl
            \setlength{\@tempdima}{\useplength{lfoldmarkhpos}}%
            \addtolengthplength[-.5]{\@tempdima}{foldmarkthickness}%
            \hskip\@tempdima
            \usekomafont{foldmark}{%
              \rule{\useplength{foldmarkthickness}}%
                   {\useplength{lfoldmarklength}}%
            }%
          }%
        }%
        \ignorespaces
      \fi
    \fi
  \endgroup
}
\newcommand*{\@foldmarks}{%
  \if@fold%
    \typeout{Foldmarks: yes}%
    \move@topt
    \ifdim \useplength{pfoldmarklength}=\z@
      \@setplength{pfoldmarklength}{4mm}%
    \fi
    \ifdim \useplength{tfoldmarklength}=\z@
      \@setplength{tfoldmarklength}{2mm}%
    \fi
    \ifdim \useplength{mfoldmarklength}=\z@
      \@setplength{mfoldmarklength}{2mm}%
    \fi
    \ifdim \useplength{bfoldmarklength}=\z@
```

```
      \@setplength{bfoldmarklength}{2mm}%
    \fi
    \ifdim \useplength{lfoldmarklength}=\z@
      \@setplength{lfoldmarklength}{4mm}%
    \fi
    \begingroup
      \@expandtwoargs\in@{P}{\@usefolds}%
      \ifin@
        \@foldmark{.5\paperheight}{\useplength{pfoldmarklength}}%
      \fi
    \endgroup
    \@hfoldmark{T}%
    \@hfoldmark{M}%
    \@hfoldmark{B}%
    \@vfoldmark{L}%
    \move@frompt
  \else
    \typeout{Foldmarks: no}%
  \fi%
}
```

`\@savevbox` `\if@savevbox@warning` `\@savevbox@warningtrue` `\@savevbox@warningfalse`

Wenn eine der vertikalen Boxen in `\opening` zu einer `overbox` `\vbox` Meldung führt, weiss letztlich niemand, welche Box dafür verantwortlich war. Deshalb definieren wir hier ein Makro, das ggf. eine Meldung ausgibt, die erklärt, welches Element die Ursache war und was man tun kann. Das erste Argument ist dabei die Sollhöhe der Box, das zweite Argument ist der Boxinhalt, das dritte Argument is der Name des Elements, das zu hoch ist, das vierte der Name der Elementhöhe und das fünfte schließlich der Befehl, mit dem das Element vom Anwender definiert werden kann. Das optionale Argument wird nur bei der Ausgabe mit ausgegeben, findet jedoch bei der Berechnung keine Berücksichtigung.

```
\newif\if@savevbox@warning\@savevbox@warningtrue
\newcommand{\@savevbox}[6][]{%
  \begingroup%
    \vbox to #2{#1#3}%
    \if@savevbox@warning\@savevbox@warningfalse%
    \setbox0\vbox{#3}%
    \setlength{\@tempdima}{#2}%
    \setlength{\@tempdimb}{\ht0}\addtolength{\@tempdimb}{\dp0}%
    \ifdim \@tempdimb>\@tempdima%
      \addtolength{\@tempdimb}{-\@tempdima}%
      \ClassWarning{%
        scrlttr2%
      }{%
        #4 is \the\@tempdimb\space too high.\MessageBreak
        You have to change `\string#5'\MessageBreak
        or you have to define a smaller\MessageBreak
        #4 using \string#6.\MessageBreak
```

```
        Because of this too high #4\MessageBreak
        you've got an overfull \string\vbox\space message%
      }%
    \fi%
    \fi%
  \endgroup%
}
```

\@firstheadfootfield

Ab Version 2.8n werden Kopf und Fuß der ersten Seite nicht mehr per Seitenstil `firstpage` gesetzt, sondern mit einem speziellen Feld. Dabei wird auch beachtet, dass dieses Feld im Kopf eine bestimmte aber beschränkte Höhe hat. Diese Höhe ist unabhängig von der normalen Kopfhöhe und richtet sich stattdessen nach der Position des Adressfeldes.

```
\newcommand*{\@firstheadfootfield}{%
  \if@firsthead
    \typeout{Head of first page}%
    \move@topt\vskip\useplength{firstheadvpos}%
    \vb@t@z{%
      \rlap{\move@topl
```

Wenn `firstheadhpos` positiv und kleiner als die Seitenbreite ist, dann ist es der Abstand des Kopfes vom linken Rand. Wenn `firstheadhpos` negativ und betragsmäßig kleiner als die Seitenbreite ist, dann ist es der Abstand des Kopfes vom rechten Rand. Wenn `firstheadhpos` größer als die Seitenbreite ist, dann wird der Kopf zentriert. Wenn `firstheadhpos` kleiner als die negative Seitenbreite ist, dann wird der Kopf am linken Rand des Satzspiegels ausgerichtet.

```
        \ifdim\useplength{firstheadhpos}<\paperwidth
          \ifdim \useplength{firstheadhpos}>-\paperwidth
            \ifdim \useplength{firstheadhpos}<\z@
              \setlength\@tempskipa{\paperwidth}%
              \addtolengthplength{\@tempskipa}{firstheadhpos}%
              \addtolengthplength[-]{\@tempskipa}{firstheadwidth}%
            \else
              \setlength{\@tempskipa}{\useplength{firstheadhpos}}%
            \fi
          \else
            \setlength\@tempskipa{\oddsidemargin}%
            \addtolength\@tempskipa{1in}%
          \fi
        \else
          \setlength\@tempskipa{.5\paperwidth}%
          \addtolengthplength[-.5]{\@tempskipa}{firstheadwidth}%
        \fi
        \hskip\@tempskipa
        \vbox{\hsize\useplength{firstheadwidth}%
          \setlength{\@tempskipa}{\useplength{toaddrvpos}}%
          \addtolengthplength[-1]{\@tempskipa}{firstheadvpos}%
          \addtolength{\@tempskipa}{-\baselineskip}%
```

Da sich einige – beispielsweise `briefkopf.sty` – darauf verlassen, dass `\textwidth` die Breite des Briefkopfes ist, wird diese Länge lokal entsprechend gesetzt. Außerdem wird dafür gesorgt, dass `\parfillskip` hier nicht berücksichtigt wird.

```
          \setlength{\textwidth}{\useplength{firstheadwidth}}%
          \setlength{\parfillskip}{\z@ \@plus 1fil}%
          \@savevbox{\@tempskipa}{\noindent\usekomavar{firsthead}%
            \vskip \z@ \@plus .00001fil}%
          {head of first page}{firstheadvpos}{\setkomavar}%
        }%
      }%
    }%
    \vskip-\useplength{firstheadvpos}\move@frompt
  \else
    \typeout{No head at first page}%
  \fi
  \move@topt
  \if@firstfoot
    \typeout{Foot of first page}%
    \vb@t@z{%
      \rlap{\move@topl
```

Wenn `firstfoothpos` positiv und kleiner als die Seitenbreite ist, dann ist es der Abstand des Kopfes vom linken Rand. Wenn `firstfoothpos` negativ und betragsmäßig kleiner als die Seitenbreite ist, dann ist es der Abstand des Kopfes vom rechten Rand. Wenn `firstfoothpos` größer als die Seitenbreite ist, dann wird der Kopf zentriert. Wenn `firstfoothpos` kleiner als die negative Seitenbreite ist, dann wird der Kopf am linken Rand des Satzspiegels ausgerichtet.

```
        \ifdim\useplength{firstfoothpos}<\paperwidth
         \ifdim \useplength{firstfoothpos}>-\paperwidth
           \ifdim \useplength{firstfoothpos}<\z@
             \setlength\@tempskipa{\paperwidth}%
             \addtolengthplength{\@tempskipa}{firstfoothpos}%
             \addtolengthplength[-]{\@tempskipa}{firstfootwidth}%
           \else
             \setlength{\@tempskipa}{\useplength{firstfoothpos}}%
           \fi
         \else
           \setlength\@tempskipa{\oddsidemargin}%
           \addtolength\@tempskipa{1in}%
         \fi
       \else
         \setlength\@tempskipa{.5\paperwidth}%
         \addtolengthplength[-.5]{\@tempskipa}{firstfootwidth}%
       \fi
       \hskip\@tempskipa
       \vbox{\hsize\useplength{firstfootwidth}%
```

```
          \vskip\useplength{firstfootvpos}%
```

Da sich einige – beispielsweise `briefkopf.sty` – darauf verlassen, dass `\textwidth` die Breite des Brieffußes ist, wird diese Länge lokal entsprechend gesetzt. Außerdem wird dafür gesorgt, dass `\parfillskip` hier nicht berücksichtigt wird.

```
          \setlength{\textwidth}{\useplength{firstfootwidth}}%
          \setlength{\parfillskip}{\z@ \@plus 1fil}%
          \vbox{\hsize\useplength{firstfootwidth}\noindent
            \usekomavar{firstfoot}}%
        }%
      }%
    }%
  \else
    \typeout{No foot at first page}%
  \fi
  \move@frompt%
  {%
    \setlength{\@tempdima}{1in}\addtolength{\@tempdima}{\topmargin}%
    \addtolength{\@tempdima}{\headheight}%
    \addtolength{\@tempdima}{\headsep}%
    \addtolength{\@tempdima}{\textheight}%
    \addtolength{\@tempdima}{\footskip}%
    \if@firstfoot
      \addtolengthplength[-1]{\@tempdima}{firstfootvpos}%
    \else
      \addtolength{\@tempdima}{-\paperheight}%
    \fi
    \ifdim\@tempdima>\z@\enlargethispage{-\@tempdima}%
    \else\if@enlargefp\enlargethispage{-\@tempdima}\fi\fi
  }%
}
```

`\@addrfield` Dasselbe gilt auch für das Adressfeld für Fensterumschläge. Die Rücksendeadresse und die Briefart werden dabei mit eigenen Formatieranweisungen gesetzt.

```
\newcommand*{\@addrfield}{%
  \ifcase \@afieldmode\relax
    \typeout{Address: No}%
  \or
    \begingroup%
      \if@baddr
        \ifkomavarempty{backaddress}{%
          \let\@tempa\@empty\let\@tempb\@empty
        }{%
          \def\@tempa{backaddress}\def\@tempb{,\space}%
        }%
      \else
        \let\@tempa\@empty\let\@tempb\@empty
      \fi
```

```
    \ifcase \ltr@priority
      \ifkomavarempty{specialmail}{}{%
        \edef\@tempa{\@tempa\@tempb specialmail}\def\@tempb{,\space}%
      }%
    \or
      \edef\@tempa{\@tempa\@tempb B-Economy}\def\@tempb{,\space}%
    \or
      \edef\@tempa{\@tempa\@tempb A-Priority}\def\@tempb{,\space}%
    \fi
    \typeout{Address (\@tempa\@tempb addressee)}%
  \endgroup%
  \move@topt\vskip\useplength{toaddrvpos}%
  \vb@t@z{{\setparsizes{\z@}{\z@}{\z@ plus 1fil}\par@updaterelative
      \rlap{\move@topl
        \setlengthtoplength{\@tempskipa}{toaddrhpos}%
        \ifdim \@tempskipa<\z@
          \addtolength{\@tempskipa}{\paperwidth}%
          \addtolengthplength[-]{\@tempskipa}{toaddrwidth}%
        \fi
        \hskip\@tempskipa
        \@savevbox{\useplength{toaddrheight}}{\hsize\useplength{toaddrwidth}%
          \@savevbox[\vss]{\useplength{backaddrheight}}{%
            \hsize\useplength{toaddrwidth}%
            \if@baddr
              \ifkomavarempty{backaddress}{}{{%
                  \noindent%
                  \def\\{\usekomavar{backaddressseparator}\@ogobble}%
                  \backaddr@format{\scriptsize\usekomafont{backaddress}%
                    \usekomavar{backaddress}}}}%
            \fi
          }{backaddress}{backaddrheight}{\setkomavar}%
          \ifcase \ltr@priority
          \or
            \setkomavar{specialmail}{%
              {\usekomafont{priority}{%
                  {\usekomafont{prioritykey}{B}}%
                  -ECONOMY}}%
            }%
          \or
            \setkomavar{specialmail}{%
              {\usekomafont{priority}{%
                  {\usekomafont{prioritykey}{A}}%
                  -PRIORITY}}%
            }%
          \fi
          \ifkomavarempty{specialmail}{}{%
            \vbox{\hsize\useplength{toaddrwidth}%
              \noindent\hspace*{\useplength{specialmailindent}}%
              \specialmail@format{%
```

```
                  \usekomafont{specialmail}\usekomavar{specialmail}}%
                \hspace*{\useplength{specialmailrightindent}}}%
            }%
            \vfil%
            \noindent\hskip\useplength{toaddrindent}%
            \setlength{\@tempdima}{\useplength{toaddrwidth}}%
            \addtolengthplength[-2]{\@tempdima}{toaddrindent}%
            \vbox{\hsize\@tempdima\raggedright
              \usekomafont{addressee}{%
                \usekomafont{toname}{\usekomavar{toname}\\}%
                \usekomafont{toaddress}{\usekomavar{toaddress}}}}%
            \vfil\null\vskip-\baselineskip}%
          {address field}{toaddrheight}%
          {\begin{letter} or you\MessageBreak
            have to define a smaller special purpose using\MessageBreak
            \string\specialmail}%
        }}}%
    \vskip-\useplength{toaddrvpos}\move@frompt
  \else
    \begingroup%
```

In den Modi `PP`, `backgroundimage` oder `addresseeimage` gibt es weder Rücksendeadresse noch Versandart aber ebenfalls eine Priorität.

```
      \if@baddr
        \let\@tempa\@empty\let\@tempb\@empty
        \ifcase \@afieldmode\relax\or\or
          \def\@tempa{PP-mode}\def\@tempb{,\space}%
        \or
          \def\@tempa{backgroundimage-mode}\def\@tempb{,\space}%
        \or
          \def\@tempa{image-mode}\def\@tempb{,\space}%
        \fi
        \ifkomavarempty{backaddress}{}{%
          \edef\@tempa{\@tempa\@tempb backaddress ignored}\def\@tempb{,\space}%
        }%
      \else
        \let\@tempa\@empty\let\@tempb\@empty
      \fi
      \ifcase \ltr@priority
        \ifkomavarempty{specialmail}{}{%
          \edef\@tempa{\@tempa\@tempb specialmail ignored}\def\@tempb{,\space}%
        }%
      \or
        \edef\@tempa{\@tempa\@tempb B-Economy}\def\@tempb{,\space}%
      \or
        \edef\@tempa{\@tempa\@tempb A-Priority}\def\@tempb{,\space}%
      \fi
      \typeout{Address (\@tempa\@tempb addressee)}%
    \endgroup%
    \move@topt\vskip\useplength{toaddrvpos}%
```

```
    \vb@t@z{{%
        \setparsizes{\z@}{\z@}{\z@ plus 1fil}\par@updaterelative
        \ifdim \useplength{toaddrindent}=\z@
          \@setplength{toaddrindent}{8mm}%
        \fi
        \ifdim \useplength{PPheadheight}=\z@
          \@setplength{PPheadheight}{20.74pt}%
        \fi
        \ifdim \useplength{PPheadwidth}=\z@
          \@setplength{PPheadwidth}{42mm}%
        \fi
        \ifdim \useplength{PPdatamatrixvskip}=\z@
          \@setplength{PPdatamatrixvskip}{9mm}%
        \fi
        \rlap{\move@topl
          \setlengthtoplength{\@tempskipa}{toaddrhpos}%
          \ifdim \@tempskipa<\z@
            \addtolength{\@tempskipa}{\paperwidth}%
            \addtolengthplength[-]{\@tempskipa}{toaddrwidth}%
          \fi
          \hskip\@tempskipa
          \@savevbox{\useplength{toaddrheight}}{\hsize\useplength{toaddrwidth}%
            \ifnum\@afieldmode=\tw@
```

Modus PP.

```
              \vskip 2pt\relax
              \hskip \useplength{toaddrindent}\relax
              \fbox{%
                \vbox to \dimexpr \useplength{PPheadheight}-2\fboxsep\relax{%
                  \setlength{\parindent}{0pt}%
                  \hsize \dimexpr \useplength{PPheadwidth}-2\fboxsep\relax
                  \setbox\@tempboxa\hbox{%
                    {\fontsize{20.74pt}{20.74pt}\selectfont
                      \usekomafont{PPlogo}{P.P.}}\hskip 2mm%
                  }%
                  \dp\@tempboxa\z@
                  \setlength{\@tempdima}{\hsize}%
                  \addtolength{\@tempdima}{-\wd\@tempboxa}%
                  \hbox{\box\@tempboxa
                    \vbox to \dimexpr
                    \useplength{PPheadheight}-2\fboxsep\relax{%
                      \vss\hsize\@tempdima
                      \raggedright
                      \fontsize{8pt}{9pt}\selectfont\usekomafont{PPdata}%
                      \usekomavar*{fromzipcode}\usekomavar{zipcodeseparator}%
                      \usekomavar{fromzipcode}\unskip\strut\\
                      \usekomavar{place}\unskip\strut
                      \hfill
                    }%
                  }%
```

```
                \vss
              }%
            }%
            \hskip 1mm
            \mbox{%
              \vbox to \useplength{PPheadheight}{%
                \addtolengthplength[-]{\hsize}{PPheadwidth}%
                \addtolength{\hsize}{-1mm}%
                \addtolengthplength[-2]{\hsize}{toaddrindent}%
                \setlength{\parindent}{0pt}%
                \fontsize{8pt}{9pt}\selectfont\usekomafont{PPdata}%
                \hfill\usekomavar{PPcode}
                \vss
                \ifcase \ltr@priority
                \or
                  \usekomafont{priority}{%
                    {\usekomafont{prioritykey}{B}}%
                    -ECONOMY\hss\vskip-\dp\strutbox
                  }%
                \or
                  \usekomafont{priority}{%
                    {\usekomafont{prioritykey}{A}}%
                    -PRIORITY\hss\vskip-\dp\strutbox
                  }%
                \fi
                \vss
              }%
            }%
            \vskip 1mm
            \vskip-\ht\strutbox
            \noindent\hskip\useplength{toaddrindent}%
            \hrulefill%
            \hskip\useplength{toaddrindent}%
            \vbox to\z@{\hsize\z@
              \vskip \useplength{PPdatamatrixvskip}\relax
              \llap{\usekomavar{PPdatamatrix}\hskip\useplength{toaddrindent}}%
              \vss
            }%
          \else
```

Modus `backgroundimage` oder `image`.

```
            \vskip 1mm
            \noindent\expandafter\hskip\useplength{toaddrindent}%
            \vbox to \useplength{PPheadheight}{%
              \addtolengthplength[-2]{\hsize}{toaddrindent}%
              \setlength{\parindent}{0pt}%
              \usekomavar{addresseeimage}%
              \vss
            }%
            \vskip 1mm
```

```
              \vskip\dp\strutbox
            \fi
            \ifnum \@afieldmode=4 \else
              \vfil%
              \noindent\hskip\useplength{toaddrindent}%
              \setlength{\@tempdima}{\useplength{toaddrwidth}}%
              \addtolengthplength[-2]{\@tempdima}{toaddrindent}%
              \vbox{\hsize\@tempdima
                \ifkomavarempty{PPdatamatrix}{}{%
                  \addtolength{\hsize}{-2mm}%
                  \settowidth\@tempdima{\usekomavar{PPdatamatrix}}%
                  \addtolength{\hsize}{-\@tempdima}%
                }%
                \raggedright
                \usekomafont{addressee}{%
                  \usekomafont{toname}{\usekomavar{toname}\\}%
                  \usekomafont{toaddress}{\usekomavar{toaddress}}}}%
            \fi
            \vss\null}%
          {address field}{toaddrheight}%
          {\string\begin{letter}}%
        }}}%
    \vskip-\useplength{toaddrvpos}\move@frompt
  \fi
}
```

`\specialmail@format` Die Anweisung, die für das Aussehen der Versandart zuständig ist.

```
\newcommand*{\specialmail@format}[1]{}
\let\specialmail@format\underline
```

`\@ogobble` In obiger Definition wird das Makro `\@ogobble` verwendet, um einen optionalen Parameter zu vernichten, falls dieser existiert.

```
\newcommand*{\@ogobble}[1][]{}
```

`\@locfield` Das „locfield“ wird unabhängig von der Adresse gesetzt.

```
\newcommand*{\@locfield}{%
  \ifdim \useplength{locwidth}=\z@%
    \@setplength{locwidth}{\paperwidth}%
    \ifdim \useplength{toaddrhpos}>\z@
      \@addtoplength[-2]{locwidth}{\useplength{toaddrhpos}}%
    \else
      \@addtoplength[2]{locwidth}{\useplength{toaddrhpos}}%
    \fi
    \@addtoplength[-1]{locwidth}{\useplength{toaddrwidth}}%
    \if@bigloc%
      \@setplength[.66667]{locwidth}{\useplength{locwidth}}%
    \else%
      \@setplength[.5]{locwidth}{\useplength{locwidth}}%
    \fi%
```

```
\fi%
```

Je nach Absenderanordnung muss hier nun ebenfalls etwas getan werden. Der Fall 0, also linksbündiger Absender im Kopf spielt hier keine Rolle.

```
\ifcase\@alignfrom
```

Im Fall 1, also zentrierter Absender im Kopf wird jedoch das Logo hier platziert, also darf die Absenderergänzung nicht leer sein.

```
\or
  \if@logo
    \ifkomavarempty{location}{\setkomavar{location}{\null}}{}%
  \fi
```

Im Fall 2, also rechtsbündiger Absender im Kopf, ist auch nichts zu tun.

```
\or
```

Im Fall 3, also linksbündiger Absender in der Absenderergänzung wird hier eine Menge ausgegeben, also darf die Absenderergänzung nicht leer sein.

```
\or
  \ifkomavarempty{location}{\setkomavar{location}{\null}}{}%
```

Im Fall 3, also linksbündiger Absender in der Absenderergänzung wird hier eine Menge ausgegeben, also darf die Absenderergänzung nicht leer sein.

```
\or
  \ifkomavarempty{location}{\setkomavar{location}{\null}}{}%
```

Alle anderen Fälle sind wie hier ebenfalls ohne Bedeutung.

```
\fi
\ifkomavarempty{location}{%
  \typeout{Location field: empty}%
  }{%
  \typeout{Location field: yes}%
  \ifdim \useplength{lochpos}=\z@%
    \@setplength{lochpos}{\useplength{toaddrhpos}}%
  \fi
  \ifdim \useplength{locvpos}=\z@%
    \@setplength{locvpos}{\useplength{toaddrvpos}}%
  \fi
  \ifdim \useplength{locheight}=\z@%
    \@setplength{locheight}{\useplength{toaddrheight}}%
  \fi
  \move@topt\vskip\useplength{locvpos}%
  \vb@t@z{\noindent\makebox[0pt][l]{%
      \move@topl
      \setlengthtoplength[-]{\@tempskipa}{lochpos}%
      \ifdim \@tempskipa<\z@
        \addtolength{\@tempskipa}{\paperwidth}%
      \else
        \addtolengthplength{\@tempskipa}{locwidth}%
      \fi
      \hskip\@tempskipa
```

```
        \makebox[0pt][r]{%
          \@savevbox{\useplength{locheight}}{%
            \noindent\hsize\useplength{locwidth}%
```

In einigen Fällen der Absenderanordnung muss hier nun eine ganze Menge getan werden. In anderen ist hingegen nur wenig zu tun. Im Fall 0, also linksbündiger Absender im Kopf wird nur die Absenderergänzung ausgegeben (das erfolgt nach der Fallunterscheidung).

```
            \ifcase\@alignfrom
```

Im Fall 1, also zentrierter Absender im Kopf wird jedoch das Logo hier platziert, die Absenderergänzung wird ggf. danach ausgegeben (das erfolgt nach der Fallunterscheidung).

```
            \or
              \if@logo
                \ifkomavarempty{fromlogo}{\null}{%
                  {\raggedleft\noindent\usekomavar{fromlogo}\\
                    \vskip\parskip}\noindent}%
              \fi
```

Im Fall 2, also rechtsbündiger Absender im Kopf, wird auch nur die Absenderergänzung ausgegeben (das erfolgt nach der Fallunterscheidung).

```
            \or
```

Im Fall 3, also linksbündiger Absender in der Absenderergänzung wird hier sozusagen der linksbündige Kopf nachgebaut. Das Logo steht dabei ggf. rechtsbündig oben.

```
            \or
              \if@logo
                \ifkomavarempty{fromlogo}{}{%
                  \raggedleft\usekomavar{fromlogo}\par}%
              \fi
              \raggedright
```

Zuerst wird der Name gesetzt. Dabei soll eine eigene Schriftart für den Namen verwendet werden.

```
              {\usekomafont{fromname}\strut\ignorespaces
                \usekomavar{fromname}%
```

Hat `\@rulefrom` den nummerischen Wert 1, so soll direkt auf den Absender eine Linie folgen. Die Länge dieser Linie kann vom Anwender bestimmt oder automatisch gewählt werden. Es werden dabei die Pseudo-Längen `fromrulewidth` und `fromrulethickness` verwendet. Ist `fromrulewidth` 0, so wird sie automatisch bestimmt. Bei der automatischen Längenbestimmung wird einfach eine Linie über die Gesamtbreite der Absenderergänzung gesetzt.

```
                {%
                  \ifnum\@rulefrom=1\\[-.5\baselineskip]%
                    \usekomafont{fromrule}{%
                      \setlength{\@tempdima}{\useplength{fromrulewidth}}%
```

```
                    \ifdim\@tempdima=\z@
                      \setlength{\@tempdima}{\useplength{locwidth}}%
                    \fi
                    \expandafter\rule\expandafter{\the\@tempdima}{%
                      \useplength{fromrulethickness}}%
                  }%
                \fi
              }%
              \\
            }%
```

Als Nächstes wird die Absenderadresse einschließlich aller optionalen Felder gesetzt. Dabei werden optionale Felder nur gesetzt, wenn der entsprechende Schalter *true* ist. Die Adresse wird ebenfalls innerhalb einer Gruppe in einer eigenen Schriftart gesetzt.

```
            {\usekomafont{fromaddress}\strut\ignorespaces
              \usekomavar{fromaddress}%
              \if@phone\\\usekomavar*{fromphone}\usekomavar{fromphone}\fi%
              \if@fax\\\usekomavar*{fromfax}\usekomavar{fromfax}\fi%
              \if@email\\\usekomavar*{fromemail}\usekomavar{fromemail}\fi%
              \if@www\\\usekomavar*{fromurl}\usekomavar{fromurl}\fi%
```

Wie weiter oben erwähnt, kann auch an dieser Stelle eine Linie gesetzt werden.

```
              \ifnum\@rulefrom=2\\[-.5\baselineskip]%
                \usekomafont{fromrule}{%
                  \setlength{\@tempdima}{\useplength{fromrulewidth}}%
                  \ifdim\@tempdima=\z@
                    \setlength{\@tempdima}{\useplength{locwidth}}%
                  \fi
                  \expandafter\rule\expandafter{\the\@tempdima}{%
                    \useplength{fromrulethickness}}}%
              \fi
              \\
            }%
```

Damit ist die Definition des linksbündigen Absenders in der Absenderergänzung abgeschlossen. Im Fall 4, also rechtsbündiger Absender in der Absenderergänzung wird hier exakt wie für Fall 3 verfahren. Nur ist jetzt alles rechtsbündig.

```
          \or
            \raggedleft
            \if@logo
              \ifkomavarempty{fromlogo}{}{\usekomavar{fromlogo}\par}%
            \fi
            {\usekomafont{fromname}\strut\ignorespaces
              \usekomavar{fromname}%
              {%
                \ifnum\@rulefrom=1\\[-.5\baselineskip]%
                  \usekomafont{fromrule}{%
```

```
                        \setlength{\@tempdima}{\useplength{fromrulewidth}}%
                        \ifdim\@tempdima=\z@
                          \setlength{\@tempdima}{\useplength{locwidth}}%
                        \fi
                        \expandafter\rule\expandafter{\the\@tempdima}{%
                          \useplength{fromrulethickness}}%
                      }%
                    \fi
                  }%
                  \\
                }%
                {\usekomafont{fromaddress}\strut\ignorespaces
                  \usekomavar{fromaddress}%
                  \if@phone\\\usekomavar*{fromphone}\usekomavar{fromphone}\fi%
                  \if@fax\\\usekomavar*{fromfax}\usekomavar{fromfax}\fi%
                  \if@email\\\usekomavar*{fromemail}\usekomavar{fromemail}\fi%
                  \if@www\\\usekomavar*{fromurl}\usekomavar{fromurl}\fi%
                  \ifnum\@rulefrom=2\\[-.5\baselineskip]%
                    \usekomafont{fromrule}{%
                      \setlength{\@tempdima}{\useplength{fromrulewidth}}%
                      \ifdim\@tempdima=\z@
                        \setlength{\@tempdima}{\useplength{locwidth}}%
                      \fi
                      \expandafter\rule\expandafter{\the\@tempdima}{%
                        \useplength{fromrulethickness}}}%
                  \fi
                  \\
                }%
```

In allen anderen Fällen wird ebenfalls die Absenderergänzung gesetzt.

```
              \fi
              \usekomavar{location}%
```

Zum Schluss wird noch verhindert, dass wegen nicht vollen vertikalen Boxen gemotzt wird.

```
              \vskip \z@ \@plus .00001fil}%
            {location field}{locheight}{\setkomavar}%
          }}%
      }%
      \vskip-\useplength{locvpos}\move@frompt
    }%
}
```

`\@datefield` Je nachdem, welche Felder definiert sind, muss eine geschäftsmäßiger Standardbrief oder ein eher persönliches Aussehen gewählt werden.

`\@set@reffield` Einzelne Felder werden mit Hilfe dieses Makros gesetzt. Als Parameter wird der Name des Feldes übergeben. Es wird dann davon ausgegangen, dass der Name des Feldes in der jeweiligen Sprache im Makro `\#1name` und der Wert des Feldes in `\var#1` zu finden ist.

```
\newcommand*{\@set@reffield}[1]{%
  \ifkomavarempty{#1}{}{%
    \settowidth{\@tempdima}{\strut\usekomafont{refvalue}\usekomavar{#1}}%
    \setbox0\hbox{\strut\usekomafont{refname}\usekomavar*{#1}}%
    \ifdim \@tempdima<\wd0\setlength{\@tempdima}{\wd0}\fi
    \if@tempswa\enskip\hfill\fi
    \parbox[t]{\@tempdima}{\noindent\raggedright
        \strut\box0\\\usekomafont{refvalue}\strut\usekomavar{#1}}%
    \@tempswatrue
  }%
}
```

`\@set@reffields` Dieses Makro enthält alle Aufrufe von `\@set@reffield`, die durchzuführen sind. Das Makro kann mit Hilfe von `\addtoreffields` erweitert werden.

```
\newcommand*{\@set@reffields}{}%
```

`\addtoreffields` Dieses Makro erwartet als obligatorisches Argument den Namen einer KOMA-Script Variable. Der Befehl arbeitet lokal zur aktuellen Umgebung.

```
\newcommand*{\addtoreffields}[1]{%
  \l@addto@macro{\@set@reffields}{\@set@reffield{#1}}%
}
```

`\removereffields` Es gibt auch einen Befehl, um alle Felder aus der Geschäftszeile zu entfernen.

```
\newcommand*{\removereffields}{%
  \renewcommand*{\@set@reffields}{}%
}
```

`\defaultreffields` Ebenso ein Befehl, um die üblichen Felder zu setzen.

```
\newcommand*{\defaultreffields}{%
  \removereffields
  \addtoreffields{yourref}%
  \addtoreffields{yourmail}%
  \addtoreffields{myref}%
  \addtoreffields{customer}%
  \addtoreffields{invoice}%
}
```

Genau diesen Befehl rufen wir auch gleich auf:

```
\defaultreffields
```

```
\newcommand*{\@datefield}{{%
    \setparsizes{\z@}{\z@}{\z@ plus 1fil}\par@updaterelative
    \ifdim\useplength{refwidth}=\z@
      \if@refwide
        \@setplength{refwidth}{\paperwidth}%
        \ifdim \useplength{toaddrhpos}>\z@
          \@setplength{refhpos}{\useplength{toaddrhpos}}%
```

```
        \else
          \@setplength[-]{refhpos}{\useplength{toaddrhpos}}%
        \fi
        \@addtoplength[-2]{refwidth}{\useplength{refhpos}}%
      \else
        \@setplength{refwidth}{\textwidth}%
        \@setplength{refhpos}{\oddsidemargin}%
        \@addtoplength{refhpos}{1in}%
      \fi
    \else\ifdim\useplength{refhpos}=\z@
        \begingroup
```

Wir bestimmen zunächst, um wieviel die Geschäftszeile schmaler ist als der Textbereich.

```
          \setlength\@tempdima{\textwidth}%
          \addtolengthplength[-]{\@tempdima}{refwidth}%
```

Dann wird bestimmt, wie groß der linke Rand tatsächlich ist.

```
          \setlength\@tempdimb{\oddsidemargin}%
          \addtolength\@tempdimb{1in}%
```

Ebenso wird bestimmt, wie groß der linke und rechte Rand zusammen sind.

```
          \setlength{\@tempdimc}{\paperwidth}%
          \addtolength{\@tempdimc}{-\textwidth}%
```

Jetzt wird aus dem Gesamtrand der Geschäftszeile der linke Rand der Geschäftszeile so bestimmt, dass das Randverhältnis dabei unverändert bleibt. Dabei erlauben wir uns eine gewisse Rechenungenauigkeit beim Verhältnis.

```
          \divide\@tempdimb by 32768\relax
          \divide\@tempdimc by 32768\relax
          \ifdim\@tempdimc=\z@\else
            \multiply\@tempdima by\@tempdimb
            \divide\@tempdima by\@tempdimc
          \fi
          \edef\@tempa{\noexpand\endgroup
            \noexpand\@setplength{refhpos}{\the\@tempdima}}%
        \@tempa
    \fi\fi
    \@tempswafalse
    \move@topt\vskip\useplength{refvpos}%
```

Jetzt erfolgt noch die Absatzeinzugkorrektur. Danach kommt die eigentliche Arbeit.

```
    \rlap{\noindent\move@topl\hskip\useplength{refhpos}%
      \vbox{\hsize\useplength{refwidth}%
        \noindent\@set@reffields%
```

Seit Version 2.97c wird hier auch noch getestet, ob die Geschäftszeile möglicherweise mit zu großer Schrift gesetzt wird.

```
        \if@tempswa
```

```
          \setbox0\hbox{\usekomafont{refname}\strutbox}%
          \ifdim\ht0>\ht\strutbox
            \ClassWarningNoLine{scrlttr}{%
              Font element ‘refname’ should not be larger than
              \string\normalsize!}%
          \fi
          \setbox0\hbox{\usekomafont{refvalue}\strutbox}%
          \ifdim\ht0>\ht\strutbox
            \ClassWarningNoLine{scrlttr}{%
              Font element ‘refvalue’ should not be larger than
              \string\normalsize!}%
          \fi
        \fi
        \ifkomavarempty{date}{%
          \if@tempswa
            \typeout{Reference line(s): with fields but without date}%
          \else
            \typeout{Reference line(s): no}%
            \aftergroup\aftergroup\aftergroup\@tempswatrue
          \fi
          }{%
          \if@tempswa
            \typeout{Reference line(s): with fields and date}%
            \settowidth{\@tempdima}{\strut
              \usekomafont{refvalue}\usekomavar{date}}%
            \setbox0\hbox{\usekomafont{refname}\usekomavar*{date}}%
            \ifdim \@tempdima<\wd0\setlength{\@tempdima}{\wd0}\fi
            \enskip\hfill
            \parbox[t]{\@tempdima}{\noindent\raggedright%
              \strut\box0\\\usekomafont{refvalue}\strut\usekomavar{date}}%
          \else
            \typeout{Reference line: only place and date}%
            \enskip\hfill
            \ifkomavarempty{place}{}{%
              \usekomavar{place}\usekomavar{placeseparator}}%
            \usekomavar{date}%
            \@tempswatrue
          \fi
          }%
        }%
      }%
    \if@tempswa\vskip-\baselineskip\fi
    \vskip\useplength{refaftervskip}%
  }%
}
```

`\@maketitle` `\@subjfield` Der eigentliche Titel wir mit `\@maketitle` gesetzt. Dies sollte jedoch ebenso wie beim Betreff (`\@subjfield`) *nicht* manuell geschehen.

```
\newcommand*{\@maketitle}{%
```

```
  \ifkomavarempty{title}{%
    \typeout{Title: no}%
  }{%
    \typeout{Title: yes}%
    {\setparsizes{\z@}{\z@}{\z@ plus 1fil}\par@updaterelative
      \centering\LARGE\usekomafont{title}\usekomavar{title}\par}%
    \vskip\baselineskip%
  }%
}
\newcommand*{\@subjfield}{{%
    \setparsizes{\z@}{\z@}{\z@ plus 1fil}\par@updaterelative
    \ifkomavarempty{subject}{%
      \typeout{Subject: no}%
    }{%
      \ifx\raggedsubject\@empty
        \ifdim\useplength{subjectvpos}=\z@
          \if@subjafter%
            \typeout{Subject: after opening}%
            \let\raggedsubject\centering%
          \else
            \typeout{Subject: before opening}%
            \def\raggedsubject{\noindent\@hangfrom}%
          \fi
        \else
          \typeout{Subject: at absolute vertical position}%
          \def\raggedsubject{\noindent\@hangfrom}%
        \fi
      \fi
      \ifdim\useplength{subjectvpos}=\z@
        \vskip\useplength{subjectbeforevskip}%
        \begingroup
      \else
        \move@topt\vskip\useplength{subjectvpos}%
        \expandafter\vb@t@z\expandafter{%
      \fi
      \usekomafont{subject}%
      \raggedsubject{\if@subj \subject@format{\strut\ignorespaces
          \usekomavar*{subject}}\fi}%
      \subject@format{\strut\ignorespaces\usekomavar{subject}}\par
      \ifdim\useplength{subjectvpos}=\z@
        \endgroup
        \vskip\useplength{subjectaftervskip}%
      \else
        }%
        \vskip-\useplength{subjectvpos}\move@frompt
      \fi
    }%
  }%
}
```

\ltr@len@subjectvpos Manchmal ist es günstiger, wenn man den Betreff absolut positionieren kann. Das geschieht über diese Pseudolänge, wenn sie *nicht* 0 ist.

```
\@newplength{subjectvpos}
\@setplength{subjectvpos}{0pt}
```

\ltr@len@subjectaftervskip Abstand nach dem Betreff. Dieser Abstand wird aus Gründen der Kompatibilität zu früheren Versionen mit 2\baselineskip voreingestellt.

```
\@newplength{subjectaftervskip}
\@setplength{subjectaftervskip}{2\baselineskip}
```

\ltr@len@subjectbeforevskip Abstand zusätzlich vor dem Betreff. Dieser Abstand wird aus Gründen der Kompatibilität zu früheren Versionen mit \baselineskip voreingestellt. Es sei darauf hingewiesen, dass hier bereits ein Abstand beispielsweise vom Titel existieren kann!

```
\@newplength{subjectbeforevskip}
\@setplength{subjectbeforevskip}{0pt}
```

20.4 Anweisungen für das Briefpapier

Für das Briefpapier gibt es etwas weniger Anweisungen. Teilweise werden hier auch die Anweisungen für den Briefbogen mit verwendet.

20.4.1 Kopf des Briefpapiers

Der Kopf des Briefpapiers wird wieder ganz normal über den Seitenstil festgelegt. Allerdings kann er entsprechende dem Briefbogen manuell definiert werden.

\nexthead Die Voreinstellung ist ebenfalls von Optionen abhängig.

\@nexthead
nexthead

```
\newcommand*{\@nexthead}{%
  \ClassWarning{scrlttr2}{%
    Deprecated usage of \string\@nexthead.\MessageBreak
    \string\usekomavar{nexthead} should be used\MessageBreak
    instead of \string\@nexthead
  }%
  \usekomavar{nexthead}%
}
\newkomavar{nexthead}
\setkomavar{nexthead}{%
  \rlap{\parbox[c]{\textwidth}{\raggedright\strut
      \ifnum\@pageat=0\pagemark\else\leftmark\fi\strut}}%
  \parbox[c]{\textwidth}{%
    \centering\strut
    \ifcase\@pageat
      \rightmark
    \or
      \pagemark
```

```
    \or
      \rightmark
    \fi\strut}%
  \llap{\parbox[c]{\textwidth}{\raggedleft\strut
      \ifcase\@pageat
        \leftmark
      \or
        \rightmark
      \or
        \pagemark
      \else
        \rightmark
      \fi\strut}}%
}%
```

Die Voreinstellung ist wiederum mit einer einfachen Benutzeranweisung änderbar.

```
\newcommand{\nexthead}{%
  \ClassWarning{scrlttr2}{%
    Deprecated usage of \string\nexthead.\MessageBreak
    You should use \string\setkomavar{nexthead}\MessageBreak
    instead of \string\nexthead
  }
  \setkomavar{nexthead}%
}
```

20.4.2 Fuß des Briefpapiers

\nextfoot
\@nextfoot
nextfoot

Je nach Optioneneinstellung kann sich die Seitenmarke in der Voreinstellung auch im Fuß befinden.

```
\newcommand*{\@nextfoot}{%
  \ClassWarning{scrlttr2}{%
    Deprecated usage of \string\@nextfoot.\MessageBreak
    \string\usekomavar{nextfoot} should be used\MessageBreak
    instead of \string\@nextfoot
  }%
  \usekomavar{nextfoot}%
}
\newkomavar{nextfoot}
\setkomavar{nextfoot}{%
  \ifnum\@pageat>2
    \parbox[c]{\textwidth}{%
      \ifcase\@pageat\or\or\or\raggedright\or\centering\or\raggedleft\fi
      \strut\pagemark%
      }%
  \fi
}
```

Auch diese Einstellung ist per Benutzeranweisung leicht zu ändern.

```
\newcommand{\nextfoot}{%
```

```
  \ClassWarning{scrlttr2}{%
    Deprecated usage of \string\nextfoot.\MessageBreak
    You should use \string\setkomavar{nextfoot}\MessageBreak
    instead of \string\nextfoot
  }
  \setkomavar{nextfoot}%
}
```

20.5 Die Brief-Umgebung

`letter` Diese Umgebung ist etwas anders definiert, als normalerweise von LaTeXgewohnt. Mit ihr können innerhalb eines Dokuments beliebig viele Briefe erzeugt werden. Eine der Abweichungen von der Standardklasse besteht darin, dass als optionales Argument Optionen gesetzt werden können.

```
\newenvironment{letter}[2][]{%
```

Als erstes wird eine neue Seite begonnen. Dies kann ggf. auch eine neue rechte Seite sein. Danach werden alle relevanten Zähler und Strafpunkte zurückgesetzt und der obligatorische Parameter ausgewertet.

```
  \newpage
  \if@twoside\ifodd\c@page\else\thispagestyle{empty}\null\newpage\fi\fi
  \setcounter{page}{\@ne}%
  \setcounter{footnote}{0}%
  \interlinepenalty=200\@processto{#2}%
```

Dann werden ggf. die übergebenen Optionen ausgeführt.

```
  \KOMAoptions{#1}%
```

Zum Schluss werden die per `\AtBeginLetter` angegeben Anweisungen ausgeführt.

```
  \@beginletterhook
}{%
  \stopletter\@@par\pagebreak\@@par%
}
```

`\stopletter` Damit man Material am Ende des Briefes einschleusen kann.

`\@processto` `\@xproc` Die Adresse, die als Argument der Briefumgebung übergeben wird, muss in Name und Adresse zerlegt werden.

```
\newcommand*{\stopletter}{}
\newcommand{\@processto}[1]{%
  \ifx\relax#1\relax\else
    \expandafter\@xproc#1\\\@nil\ifkomavarempty{toaddress}{}{%
    \expandafter\@xproc#1\@nil}%
  \fi
}
\newcommand*{\@xproc}{}
\long\def\@xproc#1\\#2\@nil{%
  \setkomavar{toname}{#1}\setkomavar{toaddress}{#2}}
```

\flushbottom Das Verhalten muss auf der letzten Briefseite wieder \raggedbottom ähneln. Das wird über \stopletter erreicht.

```
\g@addto@macro\flushbottom{%
  \g@addto@macro\stopletter{\vskip \z@\@plus.0001fil\relax}%
}
```

\AtBeginLetter
\@beginletterhook Wie oben zu sehen, wurde in den Anfang der Briefumgebung ein Haken eingefügt. Dieser wird auch tatsächlich intern genutzt. Er funktioniert prinzipiell wie der \AtBeginDocument-Haken. Zunächst macht der Haken rein gar nichts.

```
\newcommand{\AtBeginLetter}{\g@addto@macro\@beginletterhook}
\newcommand{\@beginletterhook}{}
```

\AtEndLetter Analog zu \AtBeginLetter gibt es auch ein \AtEndLetter. Als Haken wird hierbei \stopletter verwendet.

```
\newcommand{\AtEndLetter}{\g@addto@macro\stopletter}
```

\stopbreaks
\startbreaks
\@nobreakvspace
\@nobreakvspacex
\@nobreakcr Innerhalb von Briefen wird der Umbruch etwas anders gehandhabt. Dadurch soll der Brieftext besser positioniert werden.

```
\newcommand*{\stopbreaks}{\interlinepenalty \@M
 \def\par{\@@par\nobreak}\let\\=\@nobreakcr
 \let\vspace\@nobreakvspace}
\newcommand*{\@nobreakvspace}{%
  \@ifstar{\@nobreakvspacex}{\@nobreakvspacex}}
\newcommand*{\@nobreakvspacex}[1]{\ifvmode\nobreak\vskip #1\relax\else
 \@bsphack\vadjust{\nobreak\vskip #1}\@esphack\fi}
\newcommand*{\@nobreakcr}{%
  \vadjust{\penalty\@M}\@ifstar{\@xnewline}{\@xnewline}}
\newcommand*{\startbreaks}{\let\\=\@normalcr
 \interlinepenalty 200\def\par{\@@par\penalty 200}}
```

\opening Denn beim Eröffnungsgruß mit \opening wird neben sämtlichen Feldern auch der Titel und der Betreff gesetzt.

```
\newcommand{\opening}[1]{%
  \@PapersizeWarning
  \@gen@firsthead
  \@mkboth{\scr@fromname@var}{\scr@subject@var}%
  \thispagestyle{empty}\noindent
  \@foldmarks\@firstheadfootfield\@addrfield\@locfield
  \ifdim\useplength{subjectvpos}=\z@\else\@subjfield\fi
  \@datefield
  \@maketitle
  \ifdim\useplength{subjectvpos}=\z@\if@subjafter\else\@subjfield\fi\fi
  {\setparsizes{\z@}{\z@}{\z@ plus 1fil}\par@updaterelative#1\par}%
  {\setlength{\@tempdima}{\baselineskip}%
    \addtolength{\@tempdima}{-\parskip}%
    \ifdim \@tempdima>\z@\vskip\@tempdima\fi}%
  \ifdim\useplength{subjectvpos}=\z@\if@subjafter\@subjfield\fi\fi
```

```
  \@afterindentfalse\@afterheading
}
```

\closing Außer dem Eröffnungsgruß mit \opening gibt es natürlich auch einen Schlussgruß mit \closing, bei dem außerdem die Unterschrift u. ä. gesetzt wird.

\presig@skip

\setpresigskip Das Makro \presig@skip beinhaltet dabei den Abstand zwischen Schlussgruß und Signatur. Da der Wert über ein Benutzerinterface per \setpresigskip gesetzt wird, muss hier kein kostbares Längenregister (skip oder dim) verschwendet werden, sondern ein Makro reicht aus. In \setpresigskip wird allerdings noch sichergestellt, dass die Fehlermeldung bei falschem Parameter möglichst nicht erst bei Verwendung auftritt.

```
⟨/letter⟩
⟨lco & KOMAold⟩\def\setpresigskip{\@setplength{sigbeforevskip}}
⟨*letter⟩
```

\@xtrimtempa
\@@xtrimtempa Die beiden internen Macros werden benötigt, um beim Zerlegen eines mit \\[\dots] zusammengesetzten Arguments das optionale Argument in eckigen Klammern abzutrennen. Das Ergebnis steht anschließend in \@tempa.

```
\newcommand*{\@xtrimtempa}{%
  \@ifnextchar [{\@@xtrimtempa}{\@@xtrimtempa[]}}
\newcommand*{\@@xtrimtempa}{}
\def\@@xtrimtempa[#1]#2\@nil{\def\@tempa{#2}}
```

\@tempdimaismaxwidthof Das Makro zerlegt ein mit \\[\dots] zusammengesetzten Arguments und merkt in \@tempdima die Länge des längsten Elements. \@tempdima enthält beim Aufruf die minimale Länge.

```
\newcommand*{\@tempdimaismaxwidthof}{}
\def\@tempdimaismaxwidthof#1\\#2\@nil{%
  \settowidth{\@tempdimb}{\strut\ignorespaces#1}%
  \ifdim\@tempdima<\@tempdimb\setlength{\@tempdima}{\@tempdimb}\fi
  \def\@tempa{#2}\ifx\@tempa\@empty
    \let\@tempb\relax
  \else
    \expandafter\@xtrimtempa\@tempa\@nil
    \def\@tempb{\expandafter\@tempdimaismaxwidthof\@tempa\@nil}%
  \fi
  \@tempb
}
```

Jetzt endlich zum eigentichen Befehl.

```
\newcommand*{\closing}[1]{{%
    \setparsizes{\z@}{\z@}{\z@ plus 1fil}\par@updaterelative\nobreak\par%
    \nobreak\vskip\baselineskip%
    \stopbreaks\noindent%
```

```
    \setlength{\@tempdima}{0pt}%
    \expandafter\@tempdimaismaxwidthof\scr@signature@var\\\@nil
    \expandafter\@tempdimaismaxwidthof\strut\ignorespaces#1\\\@nil
    \setlength{\@tempdimb}{\hsize}%
    \addtolength{\@tempdimb}{-\useplength{sigindent}}%
    \ifdim\@tempdima>\@tempdimb\setlength{\@tempdima}{\@tempdimb}\fi%
    \hspace{\useplength{sigindent}}%
    \parbox{\@tempdima}{\raggedsignature\strut\ignorespaces%
      #1\strut\\[\useplength{sigbeforevskip}]
      \strut\usekomavar{signature}\strut}}\par
  \nobreak\vskip\baselineskip%
}
```

\ps Ebenfalls möglich ist ein Postscriptum. Dieser Befehl erwartet das Postscriptum nicht als Argument, sondern schaltet zum Nachtext um.

```
\newcommand\ps{\par%
  \ifdim\parskip=\z@%
    \vskip\baselineskip%
  \fi\startbreaks\noindent\ignorespaces%
}
```

\cc Darüber hinaus gibt es noch einen Verteiler. Dieser wird sauber formatiert.

```
\newcommand*{\cc}[1]{\par%
  \ifdim\parskip=\z@%
    \vskip\baselineskip%
  \fi\noindent%
  \begingroup
    \parbox[t]{\textwidth}{%
      \ifkomavarempty{ccseparator}{}{%
        \@hangfrom{%
          \strut\usekomavar*{ccseparator}\usekomavar{ccseparator}}%
        }%
      \ignorespaces #1\strut}%
    \setlength{\parfillskip}{\z@ \@plus 1fil}\par
  \endgroup
}
```

\encl Auch Anlagen werden ordentlich gesetzt.

```
\newcommand*{\encl}[1]{\par%
  \ifdim\parskip=\z@%
    \vskip\baselineskip%
  \fi\noindent%
  \begingroup
    \parbox[t]{\textwidth}{%
      \ifkomavarempty*{enclseparator}{}{%
        \@hangfrom{%
          \strut\usekomavar*{enclseparator}\usekomavar{enclseparator}}%
        }%
```

```
      \ignorespaces #1\strut}%
    \setlength{\parfillskip}{\z@ \@plus 1fil}\par
  \endgroup
}
```

20.6 Briefspezifische Feld-Variablen

signature
place
location
backaddress
specialmail
title
fromlogo
fromzipcode
PPcode
PPdatamatrix
addresseeimage

Zunächst besitzt die Briefklasse einige Variablen, die von Haus aus nicht über einen Ausgabenamen verfügen, weil sie üblicherweise ohne Ausgabe einer Bezeichung verwendet werden. Diese Variablen haben teilweise auch keinen vordefinierten Inhalt. Der Inhalt muss erst vom Anwender gesetzt werden. Teilweise gibt es aber auch einen vordefinierten Inhalt, der von anderen Variablen abhängt.

```
\newkomavar{signature}
\setkomavar{signature}{\usekomavar{fromname}}
\newkomavar{place}
\newkomavar{location}
\newkomavar{backaddress}
\setkomavar{backaddress}{%
  \ifkomavarempty{fromname}{}{%
    \strut\ignorespaces\usekomavar{fromname}%
    \ifkomavarempty{fromaddress}{}{\\}}%
  \ifkomavarempty{fromaddress}{}{%
    \strut\ignorespaces\usekomavar{fromaddress}}%
  }
\newkomavar{specialmail}
\newkomavar{title}
\newkomavar{fromlogo}
\newkomavar{fromzipcode}
\newkomavar{PPcode}
\newkomavar{PPdatamatrix}
\newkomavar{addresseeimage}
```

placeseparator
subjectseparator
ccseparator
enclseparator
backaddressseparator
phoneseparator
faxseparator
emailseparator
urlseparator
zipcodeseparator

Daneben gibt es einige Trennzeichen. Diese haben ggf. als Ausgabenamen das Feld, für das sie als Trennzeichen dienen. Die Trennzeichen sind vordefiniert.

```
\newkomavar[\usekomavar{place}]{placeseparator}
\setkomavar{placeseparator}{,~}
\newkomavar[\subjectname]{subjectseparator}
\setkomavar{subjectseparator}{: }
\newkomavar[\ccname]{ccseparator}
\setkomavar{ccseparator}{: }
\newkomavar[\enclname]{enclseparator}
\setkomavar{enclseparator}{: }
\newkomavar{backaddressseparator}
\setkomavar{backaddressseparator}{,~}
\newkomavar[\phonename]{phoneseparator}
\setkomavar{phoneseparator}{:~}
```

```
\newkomavar[\faxname]{faxseparator}
\setkomavar{faxseparator}{:~}
\newkomavar[\emailname]{emailseparator}
\setkomavar{emailseparator}{:~}
\newkomavar[\wwwname]{urlseparator}
\setkomavar{urlseparator}{:~}
\newkomavar{zipcodeseparator}
\setkomavar{zipcodeseparator}{\,--\,}
```

fromname
fromaddress
fromemail
fromurl
frombank
toname
toaddress
fromphone
fromfax
yourmail
yourref
myref
customer
invoice
subject
date

Desweiteren gibt es einige Variable, die auch einen Ausgabenamen haben, weil dieser üblicherweise vorangestellt wird. Eine Besonderheit stellt dabei `subject` dar. Der Ausgabename beinhaltet dabei nämlich auch die Trennzeichen, die hinter dem Namen üblicherweise stehen. Diese sind wiederum ein Variable, deren Name der Ausgabename des Feldes ist. Will man also nur den Namen aber nicht die Trennzeichen ändern, so ändert man statt dem Namen von `subject` den Namen von `subjectseparator`.

Ähnlich ist es mit `toaddress` und `fromaddress`. Dort ist der Wert von `toname` bzw. `fromname` der Ausgabename. Bei `toaddress` und `fromaddress` wird davon jedoch intern kein Gebrauch gemacht. Die Variable `date` wird bereits so vordefiniert, dass auch weiterhin `\date` funktioniert.

```
\newkomavar[\headfromname]{fromname}
\newkomavar[\usekomavar{fromname}]{fromaddress}
\newkomavar[\headtoname]{toname}
\newkomavar[\usekomavar{toname}]{toaddress}
\newkomavar[\usekomavar*{phoneseparator}%
  \usekomavar{phoneseparator}]{fromphone}
\newkomavar[\usekomavar*{faxseparator}%
  \usekomavar{faxseparator}]{fromfax}
\newkomavar[\usekomavar*{emailseparator}%
  \usekomavar{emailseparator}]{fromemail}
\newkomavar[\usekomavar*{urlseparator}%
  \usekomavar{urlseparator}]{fromurl}
\newkomavar[\bankname]{frombank}
\newkomavar[\yourrefname]{yourref}
\newkomavar[\yourmailname]{yourmail}
\newkomavar[\myrefname]{myref}
\newkomavar[\customername]{customer}
\newkomavar[\invoicename]{invoice}
\newkomavar[\usekomavar*{subjectseparator}%
  \usekomavar{subjectseparator}]{subject}
\newkomavar[\datename]{date}
\setkomavar{date}{\@date}
```

Datei u scrkvars.dtx

21 KOMA-Script-Variablen

Bei Briefen gibt es eine ganze Reihe von speziellen Variablen. An dieser Stelle sei als Beispiel nur das Feld „Ihr Zeichen“ genannt. Es ist nahezu unmöglich alle diese Felder vorherzusehen. Daher wurde in der neuen Briefklasse ein neuer Mechanismus zur Definition solcher Felder geschaffen.

21.1 Option

Der Mechanismus ist unabhängig von Optionen.

21.2 Anweisungen zum Anlegen und Verwenden von Variablen

`\newkomavar` Es wird zwischen zwei Arten von Feldern unterschieden:

- Felder für die Geschäftszeile
- Felder für sonstige Verwendung

Felder können aber auch nachträglich zur Geschäftszeile hinzugefügt werden. Die Sternvariante von `\newkomavar` führt dies jedoch automatisch durch.

`\@newkomavar` Da mit einer Sternvariante gearbeitet wird, wird ein Hilfsmakro benötigt. Das Makro erhält ein optionales und ein obligatorisches Argument. Das optionale Argument ist dabei die Voreinstellung für den *Ausgabenamen* des Feldes, der ggf. ausgegeben werden soll. Das obligatorische Argument ist *Name* des Feldes, über den das Feld angesprochen wird. Das Makro definiert zwei weitere Makros: `\scr@`*Name*`@var` und `\scr@`*Name*`@name`.

```
\newcommand*{\newkomavar}{%
  \@ifstar {\@tempswatrue\@newkomavar}{\@tempswafalse\@newkomavar}}
\newcommand*{\@newkomavar}[2][\relax]{%
  \@ifundefined{scr@#2@name}{%
    \@ifundefined{scr@#2@var}{%
      \begingroup
        \def\@tempa{#1}\def\@tempb{\relax}%
        \ifx\@tempa\@tempb\endgroup\else
        \endgroup\@namedef{scr@#2@name}{#1}%
      \fi%
      \expandafter\let\csname scr@#2@var\endcsname=\@empty
      \if@tempswa\addtoreffields{#2}\fi
    }{%
      \ClassError{%
```

```
        scrlttr2%
      }{%
        This should never happen%
      }{%
        The contents of the KOMA-Script variable ‘#2’\MessageBreak
        is undefined, but the name of the variable is
        defined.\MessageBreak
        This should never happen. So someone crashs me!%
      }%
    }%
  }{%
    \ClassError{%
      scrlttr2%
    }{%
      Variable ‘#2’ already defined%
    }{%
      I’ll ignore this command, if you’ll continue.%
    }%
  }%
}
```

`\setkomavar` Natürlich wird auch ein Befehl benötigt, um die Feldvariablen zu setzen und ihnen ggf. einen neuen Namen für die Ausgabe zu geben. Der *Name* der Feldvariable wird als obligatorisches erstes Argument übergeben. Zusätzlich zum Setzen von Ausgabename oder Wert werden ggf. auch die Hooks `\scr@`*Name*`@postsetname` bzw. `\scr@`*Name*`@postsetvar` ausgeführt.

`\@setkomavar` Da dann zwischen einer Sternvariante und einer normalen unterschieden werden soll wird ein Hilfsmakro benötigt.

`\@setkomaname` Die Sternvariante definiert nur den Ausgabename neu.

`\@@setkomavar` Da in der normalen Variante nach dem Stern ein optionales Argument folgen kann, wird mit einem weiteren Hilfsmakro gearbeitet. Der *Ausgabename* wird dabei als optionales, zweites Argument übergeben. Der *Inhalt* wird als obligatorisches, drittes Argument übergeben.

```
\newcommand*{\setkomavar}{%
  \@ifstar {\@setkomaname}{\@setkomavar}}
\newcommand*{\@setkomavar}[1]{%
  \@ifnextchar [%]
  {\@@setkomavar{#1}}{\@@setkomavar{#1}[\relax]}}
\newcommand*{\@setkomaname}[2]{%
  \@ifundefined{scr@#1@var}{%
    \@komavar@err{set}{#1}%
  }{%
    \@namedef{scr@#1@name}{#2}%
    \csname scr@#1@postsetname\endcsname
  }%
}
```

```
\newcommand*{\@@setkomavar}{}
\long\def\@@setkomavar#1[#2]#3{%
  \@ifundefined{scr@#1@var}{%
    \@komavar@err{set}{#1}%
  }{%
    \@namedef{scr@#1@var}{#3}%
    \csname scr@#1@postsetvar\endcsname
    \begingroup
      \def\@tempa{#2}\def\@tempb{\relax}%
      \ifx\@tempa\@tempb\endgroup\else
      \endgroup\@namedef{scr@#1@name}{#2}%
      \csname scr@#1@postsetname\endcsname
    \fi
  }%
}
```

`\@komavar@err` Dieses Makro wird intern in den Makros `\setkomavar` und `\usekomavar` verwendet.

```
\newcommand*{\@komavar@err}[2]{%
  \ClassError{%
    scrlttr2%
  }{%
    KOMA-Script variable not defined%
  }{%
    You've tried to #1 the not defined KOMA-Script variable
    `#2'.\MessageBreak
    You have to define the variable using \string\newkomavar\space
    before\MessageBreak
    you do this.%
  }%
}
```

`\usekomavar` Selbstverständlich wird auch ein Makro benötigt, um auf den Inhalt oder den Ausgabename einer Variablen zugreifen zu können. Mit der normalen Variante wird dabei der Inhalt, mit der Sternvariante der Name ausgegeben.

`\@usekomavar` `\@usekomaname` Da mir einer Sternvariante gearbeitet wird, werden zwei Hilfsmakros benötigt.

```
\DeclareRobustCommand*{\usekomavar}{%
  \@ifstar {\@usekomaname}{\@usekomavar}}
\newcommand*{\@usekomavar}[2][\@firstofone]{%
  \@ifundefined{scr@#2@var}{%
    \@komavar@err{use}{#2}%
    }{%
    #1{\@nameuse{scr@#2@var}}%
    }%
  }
\newcommand*{\@usekomaname}[2][\@firstofone]{%
  \@ifundefined{scr@#2@var}{%
```

```
    \@komavar@err{use}{#2}%
    }{%
    #1{\@nameuse{scr@#2@name}}%
    }%
  }
```

\ifkomavar Feststellen, ob eine KOMA-Script-Variable überhaupt definiert ist.

```
\DeclareRobustCommand*{\ifkomavar}[1]{%
  \scr@ifundefinedorrelax{scr@#1@var}{%
    \expandafter\@secondoftwo
  }{%
    \expandafter\@firstoftwo
  }%
}
```

\ifkomavarempty Es gibt auch noch ein Makro, um zu testen, ob eine KOMA-Script Variable oder deren Name leer ist. Die Sternvariante ist wieder für den Namen zuständig.

\if@komavarempty Da mit einer Sternvariante gearbeitet wird, wird wieder ein Hilfsmakro verwendet.

```
\DeclareRobustCommand*{\ifkomavarempty}{%
  \begingroup
  \@ifstar {\@tempswatrue\@ifkomavarempty}%
           {\@tempswafalse\@ifkomavarempty}%
  }
\newcommand{\@ifkomavarempty}[1]{%
    \ifkomavar{#1}{%
      \if@tempswa
        \@ifundefined{scr@#1@name}{%
          \aftergroup\@firstoftwo
        }{%
          \expandafter\ifx\csname scr@#1@name\endcsname\@empty
            \aftergroup\@firstoftwo
          \else
            \aftergroup\@secondoftwo
          \fi
        }%
      \else
        \expandafter\ifx\csname scr@#1@var\endcsname\@empty
          \aftergroup\@firstoftwo
        \else
          \aftergroup\@secondoftwo
        \fi
      \fi
    }{%
      \@komavar@err{use}{#1}%
      \aftergroup\@gobbletwo
    }%
```

```
  \endgroup
}
```

Datei v scrkplen.dtx

22 Pseudo-Längen für Briefe

Pseudo-Längen wurden für scrlttr2 eingeführt, um Längenregister zu sparen. Tatsächlich hat das LaTeX-Team kurz darauf entschieden, dass künftig ε-TeX die empfohlene Maschine für LaTeX sein soll, wodurch die Einführung von Pseudo-Längen nachträglich überflüssig wurde. Es gibt sie nun aber einmal und aus Gründen der Kompatiblität wird es sie für Briefe auch weiterhin geben.

22.1 Option

Das Prinzip der Pseudo-Längen wird nicht durch Optionen beeinflusst.

22.2 Anweisungen für Pseudo-Längen

Zu Berechnung der Feldpositionen werden verschiedene Dimensions-Variablen benötigt. Um jedoch nicht unnötig viele Längenregister zu verbrauchen, wird hier mit einer Pseudo-Längen-Arithmetik gearbeitet.

`\@newplength` Dieses Makro definiert eine neue Pseude-Länge und setzt sie auf den Wert 0. Eine Pseudo-Länge ist ein Makro, das eine Länge enthält.

```
\newcommand*{\@newplength}[1]{%
  \@ifundefined{ltr@len@#1}{%
    \expandafter\let\csname ltr@len@#1\endcsname=\z@%
    }{%
    \ClassError{scrlttr2}{%
      pseudo-length \expandafter\string\csname ltr@len@#1\endcsname
      already defined%
      }{%
      You tried to define a new pseudo-length using
      \string\@newplength\MessageBreak
      which is already defined. Try another name or
      better\MessageBreak
      Don't use this KOMA-Script internal macro}%
    }%
  }
```

`\useplength` Natürlich muss auf eine solche Pseudo-Länge auch wieder lesend zugeriffen werden können. Wir definieren dies hier so, dass es außerdem möglich ist, ein Vorzeichen oder einen Faktor zu verwenden. Dafür wird allerdings ε-TeX benötigt.

```
\scr@ifundefinedorrelax{dimexpr}{%
```

```
  \newcommand*{\useplength}[1]{%
    \number \csname ltx@len@#1\endcsname\relax}%
}{%
  \newcommand*{\useplength}[1]{%
    \dimexpr \csname ltr@len@#1\endcsname\relax}%
}
```

`\setlengthtoplength` Eine echte Länge soll auf das Vielfache einer Pseudo-Länge gesetzt werden. Dabei wird der Faktor als optionaler Parameter übergeben. Der erste obligatorische Parameter ist die Länge, der zweite obligatorische Parameter ist die Pseudo-Länge.

```
\newcommand*{\setlengthtoplength}[3][]{%
  \setlength{#2}{\useplength{#3}}%
  \setlength{#2}{#1#2}}
```

`\@setplength` Natürlich wird ebenfalls eine Möglichkeit benötigt, einer Pseudo-Länge einen Wert zuzuweisen. Dies geschieht unter lokaler Anwendung einer Länge, damit sichergestellt ist, dass eine Längenangabe übergeben wird. Dabei wird das zuvor definierte Makro `\setlengthtoplength` verwendet.

```
\newcommand*{\@setplength}[3][]{%
  \begingroup%
    \setlength{\@tempskipa}{#3}%
    \setlength{\@tempskipa}{#1\@tempskipa}%
    \edef\@tempa{\noexpand\endgroup%
      \noexpand\expandafter\noexpand\renewcommand\noexpand\expandafter*%
      \noexpand\csname ltr@len@#2\noexpand\endcsname{\the\@tempskipa}%
      }%
    \@tempa%
  }
```

`\addtolengthplength` Wenn wir schon dabei sind, soll statt dem Setzen auch das Addieren funktionieren. Hier ist die Erschwernis, dass der Faktor nur für die Pseudo-Länge und nicht für das Endergebnis gelten soll.

```
\newcommand*{\addtolengthplength}[3][]{%
  \begingroup%
    \setlengthtoplength[{#1}]{\@tempskipa}{#3}%
    \edef\@tempa{\endgroup%
      \noexpand\addtolength{#2}{\the\@tempskipa}}%
    \@tempa%
  }
```

`\@addtoplength` Ebenso wird ein Makro benötigt, mit dem zu einer Pseudo-Länge das Vielfache eines Wertes addiert werden kann.

```
\newcommand*{\@addtoplength}[3][]{%
  \begingroup%
    \setlength{\@tempskipa}{#3}%
    \setlength{\@tempskipa}{#1\@tempskipa}%
    \addtolengthplength{\@tempskipa}{#2}%
```

```
  \edef\@tempa{\noexpand\endgroup%
    \noexpand\expandafter\noexpand\renewcommand\noexpand\expandafter*%
    \noexpand\csname ltr@len@#2\noexpand\endcsname{\the\@tempskipa}%
    }%
  \@tempa%
}
```

Datei w scrklco.dtx

23 Optionendateien für Briefe

Bei der Briefklasse tritt eine Besonderheit ein. Hier ist es möglich, spezielle `lco`-Dateien zu definieren, die über Optionen geladen werden können. Die unterschiedlichen Brieftypen sind ein Fall für solche ***l**etter **c**lass **o**ption* Dateien.

23.1 Optionen für Letter-Class-Options

`\lco@test` Letter-Class-Optionen können auch für jede per `\documentclass` angegebene keyval-Option oder sonstige scan-Option (das sind Optionen, die nicht per `\DeclareOption` definiert wurden) geladen werden. Dabei wird die Abarbeitung der Option nicht unterbrochen. Dazu wird dieses Makro in den scan-Vorgang eingebaut

```
\newcommand*{\lco@test}{%
  \IfFileExists{\CurrentOption.lco}{%
    \expandafter\AtEndOfClass\expandafter{%
      \expandafter\LoadLetterOption\expandafter{\CurrentOption}%
    }%
    \expandafter\def\expandafter\scr@pti@nerr@r\expandafter{%
      \expandafter\def\expandafter\scr@pti@nerr@r\expandafter{%
        \scr@pti@nerr@r
      }%
    }%
  }{}%
}
```

Als Voreinstellung wird am Ende schon einmal die Datei `DIN.lco` geladen.

```
\AtEndOfClass{\LoadLetterOption{DIN}}
```

23.2 Makros für Letter-Class-Options

`\LoadLetterOption` Bei der neuen Briefklasse existiert eine Anweisung, mit der eine ***l**etter-**c**lass-**o**ption*-Datei geladen werden kann. Das Laden wird mit entsprechender Information in der `log`-Datei vermerkt. Damit in der Datei interne Makros verwendet werden können, muss der catcode von „@“ entsprechend gesetzt sein. Damit der Befehl sowohl in der Präambel als auch in weiteren Optionen-Dateien oder in der Klasse selbst verwendet werden kann, muss der catcode am Ende wieder restauriert werden. Dabei muss etwas trickreich gearbeitet werden, damit tatsächlich eine Schachtelung möglich

ist. So erfolgt die Umschaltung des catcodes immer nur beim ersten Aufruf innerhalb einer Schachtel.

```
\newcommand*{\LoadLetterOption}[1]{%
  \@ifundefined{@restore@catcode@level}%
    {\let\@restore@catcode@level=\z@}{}%
  \ifnum\@restore@catcode@level =\z@
    \ifnum\catcode`\@=11
      \let\@restore@catcode\relax
    \else
      \@tempcnta=\catcode`\@
      \edef\@restore@catcode{%
        \noexpand\catcode`\noexpand\@=\the\@tempcnta}%
      \makeatletter
    \fi
  \fi
  \@tempcnta=\@restore@catcode@level\relax
  \advance\@tempcnta by \@ne\relax
  \edef\@restore@catcode@level{\the\@tempcnta}%
  \InputIfFileExists{#1.lco}{%
    \ClassInfo{%
      scrlttr2%
      }{Letter-Class-Option `#1' loaded}%
    }{%
    \ClassError{%
      scrlttr2%
      }{%
      Letter-Class-Option file `#1.lco' not found%
      }{%
      You've told me to load the Letter-Class-Option `#1'. So I have
      to load\MessageBreak
      the file `#1.lco'. But the file isn't available.}%
    }%
  \@tempcnta=\@restore@catcode@level\relax
  \advance\@tempcnta by \m@ne\relax
  \edef\@restore@catcode@level{\the\@tempcnta}%
  \ifnum\@tempcnta =\z@
    \@restore@catcode
  \fi
  }
```

`\LetterOptionNeedsPapersize` Es besteht die Möglichkeit, in `lco`-Dateien eine bestimmte Papiergröße zu fordern. Dazu wird der Name der `lco`-Datei und die geforderte Papiergröse als Parameter übergeben. Stimmt die Papiergröße nicht überein, so wird bei `\opening` eine Warnung ausgegeben.

`\@PapersizeWarning`
`\@@PapersizeWarning`
`\LOPNP@size`
`\LOPNP@option`

Zur einfacheren Realisierung wird mit vier internen Hilfsmakros gearbeitet.

```
\newcommand*{\LOPNP@option}{}
\newcommand*{\LOPNP@size}{}
\newcommand*{\@PapersizeWarning}{%
```

```
  \begingroup%
    \edef\@tempc{\noexpand\@tempswatrue
      \noexpand\ifdim\paperwidth=\the\paperwidth\noexpand\else
      \noexpand\@tempswafalse\noexpand\fi
      \noexpand\ifdim\paperheight=\the\paperheight\noexpand\else
      \noexpand\@tempswafalse\noexpand\fi
      \noexpand\@@PapersizeWarning
      }
    \def\@tempb{letter}\ifx\LOPNP@size\@tempb%
      \setlength{\paperheight}{11in}\setlength{\paperwidth}{8.5in}%
    \else\def\@tempb{legal}\ifx\LOPNP@size\@tempb%
      \setlength{\paperheight}{14in}\setlength{\paperwidth}{8.5in}%
    \else\def\@tempb{executive}\ifx\LOPNP@size\@tempb%
      \setlength{\paperheight}{10.5in}\setlength{\paperwidth}{7.25in}%
    \else%
      \def\ta@opt@err{%
        \ClassError{%
          scrlttr2%
          }{papersize `\LOPNP@size' undefined}{%
          You've told me to check for paper size
          `\LOPNP@size'\MessageBreak
          at letter option file `\LOPNP@option.lco',\MessageBreak
          but this paper size is not supported.\MessageBreak
          See the KOMA-Script manual for informations about known
          paper sizes.}%
        }%
      \let\@headlines\ta@opt@err%
      \expandafter\x@@paper\LOPNP@size paperpaper\x@@paper%
    \fi\fi\fi
    \@tempc
  \endgroup%
}
\newcommand*{\@@PapersizeWarning}{%
  \if@tempswa\else%
  \ClassWarningNoLine{%
    scrlttr2%
    }{%
    Letter option file `\LOPNP@option.lco'\MessageBreak
    needs paper size `\LOPNP@size'.\MessageBreak
    Current paper size is not `\LOPNP@size'!\MessageBreak
    You should add `paper=\LOPNP@size' at the\MessageBreak
    option list of \string\documentclass!\MessageBreak
    Maybe you know what you are doing,\MessageBreak
    so I do not change this myself}%
  \fi
  }
\newcommand*{\LetterOptionNeedsPapersize}[2]{%
  \edef\LOPNP@size{#2}\edef\LOPNP@option{#1}%
  }
```

23.3 Die Letter-Class-Optionen

Die Letter-Class-Optionen selbst werden wie bereits erwähnt über Dateien realisiert. In den vorgefertigen Dateien für unterschiedliche Arten von Briefen werden in erster Linie Pseudo-Längen passend gesetzt. In einigen Fällen werden aber auch zusätzlich Befehle umdefiniert.

23.3.1 Der Datei Kopf

Im Kopf ist natürlich die Datei zu deklarieren:

```
\ProvidesFile{%
⟨visualize⟩  visualize%
⟨DIN⟩  DIN%
⟨DINmoretext⟩ DINmtext%
⟨SN⟩  SN%
⟨SNold⟩  SNold%
⟨SwissLeft⟩  SNleft%
⟨KOMAold⟩  KOMAold%
⟨NF⟩  NF%
⟨USletter2w⟩  USletter2w
⟨UScommercial9⟩  UScommercial9
⟨UScommercial9DW⟩  UScommercial9DW
  .lco}[\KOMAScriptVersion\space letter-class-option]
```

`\LCOWarningNoLine` `\LCOWarning`

Außerdem wird ein Makro benötigt, das analog zu `\ClassWarningNoLine` oder `\PackageWarningNoLine` funktioniert.

23.3.2 Der Hauptteil

Die Maße für einige Optionen wurden nicht von mir selbst recherchiert:

`lco`-Datei	Recherchiert von
DIN	mir
DINmtext	mir
KOMAold	mir
SNleft	Roger Luethi
SN	Roger Luethi
NF	Jean-Marie Pacquet
USletter2w	Engelbert Buxbaum
UScommercial9	mir
UScommercial9DW	mir

ll

Zunächst wird geprüft, ob überhaupt die richtige Klasse geladen ist.

```
\@ifundefined{scr@fromname@var}{%
  \LCOWarningNoLine{%
⟨DIN⟩    DIN%
⟨DINmoretext⟩   DINmtext%
```

```
⟨SN⟩     SN%
⟨SNold⟩     SNold%
⟨SwissLeft⟩     SNleft%
⟨KOMAold⟩     KOMAold%
⟨NF⟩     NF%
⟨USletter2w⟩     USletter2w%
⟨UScommercial9⟩     UScommercial9%
⟨UScommercial9DW⟩     UScommercial9DW%
  }{%
    This letter class option file was made only\MessageBreak
    to be used with KOMA-Script letter class\MessageBreak
    ‘scrlttr2’. Use with other classes may result\MessageBreak
    in a lot of errors%
  }%
}{}
```

Dann wird geprüft, ob das richtige Papierformat vorhanden ist.

```
\LetterOptionNeedsPapersize{%
⟨DIN⟩  DIN%
⟨DINmoretext⟩  DINmtext%
⟨SN⟩  SN%
⟨SNold⟩  SNold%
⟨SwissLeft⟩  SNleft%
⟨KOMAold⟩  KOMAold%
⟨NF⟩  NF%
⟨USletter2w⟩  USletter2w%
⟨UScommercial9⟩  UScommercial9%
⟨UScommercial9DW⟩  UScommercial9DW%
⟨*DIN | DINmoretext | SN | SNold | SwissLeft | KOMAold | NF⟩
}{a4}
⟨/DIN | DINmoretext | SN | SNold | SwissLeft | KOMAold | NF⟩
⟨*USletter2w | UScommercial9 | UScommercial9DW⟩
}{letter}
⟨/USletter2w | UScommercial9 | UScommercial9DW⟩
```

Dann werden die Werte gesetzt.

ZIP-Code

```
    ⟨!KOMAold⟩\setkomavar*{fromzipcode}{%
    ⟨DIN | DINmoretext⟩  D}
    ⟨SwissLeft | SN⟩  CH}
    ⟨NF⟩  F}
    ⟨USletter2w | UScommercial9 | UScommercial9DW⟩  US}
```

Faltmarken

```
    \@setplength{foldmarkhpos}{3.5mm}
    \@setplength{tfoldmarkvpos}{%
    ⟨DIN | SwissLeft | SN⟩  105mm}
    ⟨DINmoretext⟩  87mm}
    ⟨KOMAold⟩  103.5mm}
```

```
⟨NF⟩  99mm}
⟨USletter2w | UScommercial9 | UScommercial9DW⟩  3.75in}
\@setplength{mfoldmarkvpos}{\z@}
\@setplength{bfoldmarkvpos}{%
⟨DIN | SwissLeft | SN⟩  210mm}
⟨DINmoretext⟩  192mm}
⟨KOMAold⟩  202.5mm}
⟨NF⟩  198mm}
⟨USletter2w⟩  7.4in}
⟨UScommercial9 | UScommercial9DW⟩  7.375in}
\@setplength{lfoldmarkhpos}{\z@}
```

Anschrift

```
\@setplength{toaddrvpos}{%
⟨DIN | SN⟩  45mm}
⟨DINmoretext⟩  27mm}
⟨SwissLeft⟩  35.5mm}
⟨KOMAold⟩  49mm}
⟨NF⟩  35mm}
⟨USletter2w⟩  1.8in}
⟨UScommercial9 | UScommercial9DW⟩  2.1875in}
\@setplength{toaddrhpos}{%
⟨DIN | DINmoretext | SwissLeft⟩  20mm}
⟨SN⟩  -8mm}
⟨KOMAold⟩  1in}
⟨KOMAold⟩\@addtoplength{toaddrhpos}{\oddsidemargin}
⟨NF⟩  -10mm}
⟨USletter2w⟩  .73in}
⟨UScommercial9⟩  0.6875in}
⟨UScommercial9DW⟩  .5625in}
\@setplength{toaddrwidth}{%
⟨DIN | DINmoretext⟩  85mm}
⟨SN⟩  90mm}
⟨SwissLeft | NF⟩  100mm}
⟨KOMAold⟩  70mm}
⟨USletter2w⟩  3.11in}
⟨UScommercial9⟩  4.5in}
⟨UScommercial9DW⟩  3.625in}

\@setplength{toaddrheight}{%
⟨DIN | DINmoretext | SN | SwissLeft | KOMAold | NF⟩  45mm}
⟨USletter2w⟩  1.17in}
⟨UScommercial9 | UScommercial9DW⟩  1.125in}
\@setplength{toaddrindent}{%
⟨DIN | DINmoretext | SN | SwissLeft | KOMAold | USletter2w | UScommercial9 | UScommercial9
⟨NF⟩  10mm}
\@setplength{backaddrheight}{%
⟨DIN | DINmoretext | SN | SwissLeft | KOMAold | NF⟩  5mm}
⟨USletter2w | UScommercial9 | UScommercial9DW⟩  \z@}
```

```
⟨USletter2w | UScommercial9 | UScommercial9DW⟩\KOMAoptions{backaddress=false}
\@setplength{specialmailindent}{\fill}
\@setplength{specialmailrightindent}{1em}
```

Absenderergänzung

```
\@setplength{locwidth}{%
⟨DIN | DINmoretext | SN | SwissLeft | KOMAold | NF | USletter2w | UScommercial9 | UScommer
```

Kopf

```
\@setplength{firstheadvpos}{%
⟨DIN | DINmoretext | SN | SwissLeft⟩  8mm}
⟨KOMAold⟩  6mm}
⟨NF⟩  15mm}
⟨USletter2w⟩  .36in}
⟨UScommercial9 | UScommercial9DW⟩  .4375in}
\@setplength{firstheadwidth}{%
⟨KOMAold⟩  \textwidth}
⟨NF⟩  170mm}
⟨*DIN | DINmoretext | SN | SwissLeft | UScommercial9⟩
  \paperwidth}
\ifdim \useplength{toaddrhpos}>\z@
  \@addtoplength[-2]{firstheadwidth}{\useplength{toaddrhpos}}
\else
  \@addtoplength[2]{firstheadwidth}{\useplength{toaddrhpos}}
\fi
⟨/DIN | DINmoretext | SN | SwissLeft | UScommercial9⟩
⟨*USletter2w⟩
  \paperwidth}
\@addtoplength[-]{firstheadwidth}{.68in}
⟨/USletter2w⟩
⟨*UScommercial9DW⟩
  \paperwidth}
\@addtoplength[-]{firstheadwidth}{.25in}
⟨/UScommercial9DW⟩
```

Fuß

```
\@setplength{firstfootwidth}{\useplength{firstheadwidth}}
⟨KOMAold⟩\@setplength{firstfootvpos}{\paperheight}
⟨KOMAold⟩\@addtoplength{firstfootvpos}{-2cm}
⟨!KOMAold&!NF⟩\ifnum \scr@compatibility >\@nameuse{scr@v@2.9t}\relax
⟨!KOMAold&!NF⟩  \@setplength{firstfootvpos}{\paperheight}
⟨!KOMAold&!NF⟩  \@addtoplength{firstfootvpos}{-16mm}
⟨!KOMAold&!NF⟩\else
⟨!KOMAold&!NF⟩  \@setplength{firstfootvpos}{1in}
⟨!KOMAold&!NF⟩  \@addtoplength{firstfootvpos}{\topmargin}
⟨!KOMAold&!NF⟩  \@addtoplength{firstfootvpos}{\headheight}
⟨!KOMAold&!NF⟩  \@addtoplength{firstfootvpos}{\headsep}
⟨!KOMAold&!NF⟩  \@addtoplength{firstfootvpos}{\textheight}
```

```
    ⟨!KOMAold&!NF⟩  \@addtoplength{firstfootvpos}{\footskip}
    ⟨!KOMAold&!NF⟩\fi
    ⟨NF⟩\@setplength{firstfootvpos}{266.679mm}
```

Geschäftszeile

```
    \@setplength{refvpos}{%
    ⟨DIN | SN⟩  98.5mm}
    ⟨DINmoretext⟩  80.5mm}
    ⟨SwissLeft⟩  89mm}
    ⟨KOMAold⟩  89.5mm}
    ⟨NF⟩  \useplength{tfoldmarkvpos}}
    ⟨USletter2w⟩  3.1in}
    ⟨UScommercial9 | UScommercial9DW⟩  3.4375in}
    \@setplength{refaftervskip}{%
    ⟨!KOMAold⟩  \baselineskip}
    ⟨KOMAold⟩  2\baselineskip}
```

Bei der Breite der Geschäftszeile brauchen wir wieder eine Sonderlösung, damit die Option „wide“ auch später noch gesetzt werden kann. Also verwenden wir hier den Wert 0 als mit der Bedeutung „ist bei `\opening` zu berechnen.

```
    \@setplength{refwidth}{0pt}
```

Signature

```
    \@setplength{sigindent}{0mm}
    \@setplength{sigbeforevskip}{2\baselineskip}
    ⟨DIN | DINmoretext | SN | SwissLeft⟩\let\raggedsignature=\centering
    ⟨KOMAold | NF | USletter2w | UScommercial9 | UScommercial9DW⟩\let\raggedsignature=
```

Es folgen ggf. noch zusätzliche Änderungen.

```
⟨*DINmoretext⟩
\renewcommand*{\@firsthead}{\noindent
  \parbox[b]{\useplength{firstheadwidth}}{\centering%
    \usekomafont{fromaddress}{%
      \ifkomavarempty{fromname}{}{%
        \usekomafont{fromname}%
        {\ignorespaces\usekomavar{fromname}\\[-8pt]
          \rule{\useplength{firstheadwidth}}{.4pt}\\
          }%
        }%
      \ifkomavarempty{fromaddress}{}{%
        \def\\{,\ \@ogobble}%
        \ignorespaces\usekomavar{fromaddress}%
        }%
      }%
    }%
  }
⟨/DINmoretext⟩
```

Außerdem gibt es noch die Kompatibilitätseinstellungen, die ebenfalls von `KOMAold` aktiviert werden.

```
⟨*KOMAold⟩
\def\yourref{\setkomavar{yourref}}
\def\yourmail{\setkomavar{yourmail}}
\def\myref{\setkomavar{myref}}
\def\customer{\setkomavar{customer}}
\def\invoice{\setkomavar{invoice}}
\@ifundefined{scr@refitemi@var}{\newkomavar{refitemi}}{}
\def\refitemi{\setkomavar{refitemi}}
\def\refitemnamei{\setkomavar*{refitemi}}
\@ifundefined{scr@refitemii@var}{\newkomavar{refitemii}}{}
\def\refitemii{\setkomavar{refitemii}}
\def\refitemnameii{\setkomavar*{refitemii}}
\@ifundefined{scr@refitemiii@var}{\newkomavar{refitemiii}}{}
\def\refitemiii{\setkomavar{refitemiii}}
\def\refitemnameiii{\setkomavar*{refitemiii}}
\l@addto@macro{\defaultreffields}{%
  \addtoreffields{refitemi}%
  \addtoreffields{refitemii}%
  \addtoreffields{refitemiii}%
  }
\addtoreffields{refitemi}%
\addtoreffields{refitemii}%
\addtoreffields{refitemiii}%
\def\toname{\usekomavar{toname}}
\def\toaddress{\usekomavar{toaddress}}
\@ifundefined{scr@branch@var}{\newkomavar{branch}}{}
\def\branch{\setkomavar{branch}}
\def\frombranch{\usekomavar{branch}}
\def\signature{\setkomavar{signature}}
\def\fromsig{\usekomavar{signature}}
\def\name{\setkomavar{fromname}}
\def\fromname{\usekomavar{fromname}}
\def\address{\setkomavar{fromaddress}}
\def\fromaddress{\usekomavar{fromaddress}}
\def\location{\setkomavar{location}}
\def\fromlocation{\usekomavar{location}}
\def\backaddress{\setkomavar{backaddress}}
\def\@tempa{%
  \ifkomavarempty{fromname}{}{%
    \strut\ignorespaces\usekomavar{fromname}%
    \ifkomavarempty{fromaddress}{}{\\}}%
  \ifkomavarempty{fromaddress}{}{%
    \strut\ignorespaces\usekomavar{fromaddress}}%
  }
\ifx\@tempa\scr@backaddress@var%
  \setkomavar{backaddress}{}
\fi
```

```
\def\frombackaddress{\usekomavar{backaddress}}
\def\telephone{\setkomavar{fromphone}}
\def\telephonenum{\usekomavar{fromphone}}
\def\specialmail{\setkomavar{specialmail}}
\def\@specialmail{\usekomavar{specialmail}}
\def\title{\setkomavar{title}}
\def\@title{\usekomavar{title}}
\def\subject{\setkomavar{subject}}
\def\@subject{\usekomavar{subject}}
\def\place{\setkomavar{place}}
\def\fromplace{\usekomavar{place}}
\let\ccnameseparator=\scr@ccseparator@var
\setkomavar{ccseparator}{\ccnameseparator}
\let\enclnameseparator=\scr@enclseparator@var
\setkomavar{enclseparator}{\enclnameseparator}
\setkomafont{fromname}{\scshape}
\def\foldmarkson{\@ObsoleteCommand{\foldmarkson}{foldmarks=on}}
\def\foldmarksoff{\@ObsoleteCommand{\foldmarksoff}{foldmarks=off}}
\def\addrfieldon{\@ObsoleteCommand{\addrfieldon}{addrfield=on}}
\def\addrfieldoff{\@ObsoleteCommand{\addrfieldoff}{addrfield=off}}
\def\subjecton{\@ObsoleteCommand{\subjecton}{subject=titled}}
\def\subjectoff{\@ObsoleteCommand{\subjectoff}{subject=untitled}}
\def\subjectafteron{%
  \@ObsoleteCommand{\subjectafteron}{subject=afteropening}}
\def\subjectafteroff{%
  \@ObsoleteCommand{\subjectafteroff}{subject=beforeopening}}
⟨/KOMAold⟩
```

23.4 Makros zur Visualisierung diverser Felder in einem Brief

Diese Makros werden nur von `visualize.lco` verwendet und stellen den Hauptteil dieser LCO dar.

Das Paket benötigt eso-pic. Das Paket wird nur dann explizit geladen, wenn dies nicht bereits geschehen ist. Damit ist es möglich das Paket mit Optionen vor der LCO zu laden:

```
\@ifpackageloaded{eso-pic}{}{\RequirePackage{eso-pic}}
```

`\showfields` Diese Anweisung erlaubt es, eine Reihe von Felder zu visualisieren. Dazu gibt man die gewünschten Felder als Komma separierte Liste im ersten und einzigen Argument der Anweisung an. Gibt man ein unbekanntes Feld an, so wird nicht etwa ein Fehler, sondern nur eine Warnung ausgegeben. Die Visualisierung erfolgt im Hintergrund. Dies wird mit Hilfe des Pakets eso-pic erreicht. Wichtig ist noch, dass die Anweisung nicht zurückgenommen werden kann.

```
\newcommand*{\showfields}[1]{%
  \AtBeginLetter{%
    \begingroup
      \@for \@tempa:=#1\do{%
```

```
        \ifx\@tempa\@empty\else
          \@ifundefined{showfield@\@tempa}{%
            \LCOWarning{visualize}{Unknown field `\@tempa' ignored}%
          }{%
            \edef\@tempa{%
              \noexpand\AddToShipoutPicture*{%
                \noexpand\AtPageUpperLeft{%
                  \noexpand\usekomafont{field}%
                  \noexpand\@nameuse{showfield@\@tempa}%
                }%
              }%
            }\@tempa
          }%
        \fi
      }%
    \endgroup
  }%
}
```

`field` Die Visualisierung erfolgt dabei in einer wählbaren Farbe, die über das Element `\field` definiert ist. Voreingestellt ist dabei `\normalcolor`.

```
\newkomafont{field}{\normalcolor}
```

`\showfield` Die Visualisierung der einzelnen Felder erfolgt über eigene Anweisungen des Aufbaus `\showfield@`*`Feldname`*. In der Regel sind die Felder rechteckig. Zur Visualisierung solcher Felder können daher einheitliche Anweisungen verwendet werden. Diese besitzen vier Argumente:

1. Abstand vom linken Rand (x-Koordinate des Rahmens)
2. Abstand vom oberen Rand (y-Koordinate des Rahmens)
3. Breite des Feldes
4. Höhe des Feldes; ist dieser Wert negativ gibt es keine feste Höhe, sondern nur eine Art unbestimmte Ausdehnung mit einer Mindesthöhe

`\showfield@by@frame` Diese Anweisung visualisiert ein rechteckiges Feld durch Rahmen. Bei einer negativen Höhe wird ein unten offener Rahmen gezeichnet, bei dem die vertikalen Linien als Pfeile nach unten ausgeführt werden.

```
\newcommand*{\showfield@by@frame}[4]{%
  \thinlines
  \ifdim #4<\z@
    \put(\LenToUnit{#1},-\LenToUnit{#2}){\line(1,0){\LenToUnit{#3}}}%
    \setlength{\@tempdima}{\dimexpr #1+#3\relax}%
    \setlength{\@tempdimb}{#4}
    \put(\LenToUnit{#1},-\LenToUnit{#2}){%
      \vector(0,-1){\LenToUnit{-\@tempdimb}}}%
    \put(\LenToUnit\@tempdima,-\LenToUnit{#2}){%
```

```
      \vector(0,-1){\LenToUnit{-\@tempdimb}}}%
  \else
    \put(\LenToUnit{#1},-\LenToUnit{#2}){\line(1,0){\LenToUnit{#3}}}%
    \put(\LenToUnit{#1},-\LenToUnit{#2}){\line(0,-1){\LenToUnit{#4}}}%
    \setlength{\@tempdima}{\dimexpr #1+#3\relax}%
    \setlength{\@tempdimb}{\dimexpr #2+#4\relax}%
    \put(\LenToUnit\@tempdima,\LenToUnit{-\@tempdimb}){%
      \line(-1,0){\LenToUnit{#3}}}%
    \put(\LenToUnit\@tempdima,\LenToUnit{-\@tempdimb}){%
      \line(0,1){\LenToUnit{#4}}}%
  \fi
}
```

`\showfield@by@edges` Diese Anweisung visualisert ein rechteckiges Feld, indem es die Ecken des Feldes durch kleine Striche markiert. Bei negativer Höhe entfallen die unteren Eckmarkierungen.

```
\newcommand*{\showfield@by@edges}[4]{%
  \thinlines
  \setlength{\@tempdima}{\dimexpr #1+#3\relax}%
  \put(\LenToUnit{#1},-\LenToUnit{#2}){\line(1,0){\LenToUnit{\edgesize}}}%
  \put(\LenToUnit{#1},-\LenToUnit{#2}){\line(0,-1){\LenToUnit{\edgesize}}}%
  \put(\LenToUnit{\@tempdima},-\LenToUnit{#2}){\line(-1,0){\LenToUnit{\edgesize}
  \put(\LenToUnit{\@tempdima},-\LenToUnit{#2}){\line(0,-1){\LenToUnit{\edgesize}
  \ifdim #4<\z@\else
    \setlength{\@tempdimb}{\dimexpr #2+#4\relax}%
    \put(\LenToUnit{#1},-\LenToUnit{\@tempdimb}){\line(1,0){\LenToUnit{\edgesize
    \put(\LenToUnit{#1},-\LenToUnit{\@tempdimb}){\line(0,1){\LenToUnit{\edgesize
    \put(\LenToUnit{\@tempdima},-\LenToUnit{\@tempdimb}){\line(-1,0){\LenToUnit{\
    \put(\LenToUnit{\@tempdima},-\LenToUnit{\@tempdimb}){\line(0,1){\LenToUnit{\e
  \fi
}
```

`\edgesize` Die Länge der Ecklinien wird über dieses Makro bestimmt. Voreingestellt ist 1 ex.

```
\newcommand*{\edgesize}{1ex}
```

`\showfield@by@rule` Diese Anweisung visualisiert ein rechteckiges Feld, indem es das komplette Feld einfärbt. Es wird hier nicht zwischen unten offenen und geschlossenen Feldern unterschieden.

```
\newcommand*{\showfield@by@rule}[4]{%
  \ifdim #4<\z@
    \put(\LenToUnit{#1},-\LenToUnit{#2}){\rule[#4]{#3}{-#4}}%
  \else
    \put(\LenToUnit{#1},-\LenToUnit{#2}){\rule[-#4]{#3}{#4}}%
  \fi
}
```

Die Voreinstellung für die Visualsierung ist mit Rahmen:

```
\newcommand*{\showfield}{}
\let\showfield\showfield@by@frame
```

\setshowstyle Dieser Stil der Visualisierung kann per Anweisung umgeschaltet werden. Existiert ein Stil nicht, so wird kein Fehler gemldet, sondern lediglich eine Warnung und der Standardstil verwendet:

```
\newcommand*{\setshowstyle}[1]{%
  \@ifundefined{showfield@by@#1}{%
    \LCOWarning{visualize}{Unknown show type ‘#1’.\MessageBreak
      You should simply set one of the supported\MessageBreak
      show types ‘frame’, ‘edge’, ‘rule’.\MessageBreak
      Style ‘frame’ will be used instead}%
    \let\showfield\showfield@by@frame
  }{%
    \expandafter\let\expandafter\showfield\csname showfield@by@#1\endcsname
  }%
}
```

Die folgenden Felder werden derzeit unterstützt:

\showfield@test ein Testfeld `test` mit den Koordinaten: $(1\,\mathrm{cm}, 1\,\mathrm{cm}) \times (10\,\mathrm{cm}, 15\,\mathrm{cm})$

```
\newcommand*{\showfield@test}{%
  \showfield{1cm}{1cm}{10cm}{15cm}%
}
```

\showfield@head das Feld `head` für den Kopfbereich des Briefbogens:

```
\newcommand*{\showfield@head}{%
  \ifdim\useplength{firstheadhpos}<\paperwidth
    \ifdim \useplength{firstheadhpos}>-\paperwidth
      \ifdim \useplength{firstheadhpos}<\z@
        \setlength\@tempskipa{\paperwidth}%
        \addtolengthplength{\@tempskipa}{firstheadhpos}%
        \addtolengthplength[-]{\@tempskipa}{firstheadwidth}%
      \else
        \setlength{\@tempskipa}{\useplength{firstheadhpos}}%
      \fi
    \else
      \setlength\@tempskipa{\oddsidemargin}%
      \addtolength\@tempskipa{1in}%
    \fi
  \else
    \setlength\@tempskipa{.5\paperwidth}%
    \addtolengthplength[-.5]{\@tempskipa}{firstheadwidth}%
  \fi
  \showfield{\@tempskipa}%
            {\useplength{firstheadvpos}}%
            {\useplength{firstheadwidth}}%
            {-\headheight}%
}
```

\showfield@foot das Feld foot für den Fußbereich des Briefbogens:

```
\newcommand*{\showfield@foot}{%
  \ifdim\useplength{firstfoothpos}<\paperwidth
    \ifdim \useplength{firstfoothpos}>-\paperwidth
      \ifdim \useplength{firstfoothpos}<\z@
        \setlength\@tempskipa{\paperwidth}%
        \addtolengthplength{\@tempskipa}{firstfoothpos}%
        \addtolengthplength[-]{\@tempskipa}{firstfootwidth}%
      \else
        \setlength{\@tempskipa}{\useplength{firstfoothpos}}%
      \fi
    \else
      \setlength\@tempskipa{\oddsidemargin}%
      \addtolength\@tempskipa{1in}%
    \fi
  \else
    \setlength\@tempskipa{.5\paperwidth}%
    \addtolengthplength[-.5]{\@tempskipa}{firstfootwidth}%
  \fi
  \showfield{\@tempskipa}%
            {\useplength{firstfootvpos}}%
            {\useplength{firstfootwidth}}%
            {-\baselineskip}%
}
```

\showfield@address das Feld address für das Feld mit der Anschrift:

```
\newcommand*{\showfield@address}{%
  \setlengthtoplength{\@tempskipa}{toaddrhpos}%
  \ifdim \@tempskipa<\z@
    \addtolength{\@tempskipa}{\paperwidth}%
    \addtolengthplength[-]{\@tempskipa}{toaddrwidth}%
  \fi
  \showfield{\@tempskipa}{\useplength{toaddrvpos}}%
            {\useplength{toaddrwidth}}%
            {\dimexpr\useplength{toaddrheight}\relax}%
}
```

\showfield@location das Feld location mit der Absenderergänzung:

```
\newcommand*{\showfield@location}{%
  \begingroup
    \ifdim \useplength{locwidth}=\z@%
      \@setplength{locwidth}{\paperwidth}%
      \ifdim \useplength{toaddrhpos}>\z@
        \@addtoplength[-2]{locwidth}{\useplength{toaddrhpos}}%
      \else
        \@addtoplength[2]{locwidth}{\useplength{toaddrhpos}}%
      \fi
      \@addtoplength[-1]{locwidth}{\useplength{toaddrwidth}}%
      \if@bigloc%
```

```
        \@setplength[.66667]{locwidth}{\useplength{locwidth}}%
      \else%
        \@setplength[.5]{locwidth}{\useplength{locwidth}}%
      \fi%
    \fi%
    \ifdim \useplength{lochpos}=\z@%
      \@setplength{lochpos}{\useplength{toaddrhpos}}%
    \fi
    \ifdim \useplength{locvpos}=\z@%
      \@setplength{locvpos}{\useplength{toaddrvpos}}%
    \fi
    \ifdim \useplength{locheight}=\z@%
      \@setplength{locheight}{\useplength{toaddrheight}}%
    \fi
    \setlengthtoplength[-]{\@tempskipa}{lochpos}%
    \ifdim \@tempskipa<\z@
      \addtolength{\@tempskipa}{\paperwidth}%
    \else
      \addtolengthplength{\@tempskipa}{locwidth}%
    \fi
    \addtolengthplength[-]{\@tempskipa}{locwidth}%
    \showfield{\@tempskipa}{\useplength{locvpos}}%
              {\useplength{locwidth}}{\useplength{locheight}}%
  \endgroup
}
```

`\showfield@refline` das Feld mit der Geschäftszeile:

```
\newcommand*{\showfield@refline}{%
  \begingroup
    \ifdim\useplength{refwidth}=\z@
      \if@refwide
        \@setplength{refwidth}{\paperwidth}%
        \ifdim \useplength{toaddrhpos}>\z@
          \@setplength{refhpos}{\useplength{toaddrhpos}}%
        \else
          \@setplength[-]{refhpos}{\useplength{toaddrhpos}}%
        \fi
        \@addtoplength[-2]{refwidth}{\useplength{refhpos}}%
      \else
        \@setplength{refwidth}{\textwidth}%
        \@setplength{refhpos}{\oddsidemargin}%
        \@addtoplength{refhpos}{1in}%
      \fi
    \else\ifdim\useplength{refhpos}=\z@
        \begingroup
          \setlength\@tempdima{\textwidth}%
          \addtolengthplength[-]{\@tempdima}{refwidth}%
          \setlength\@tempdimb{\oddsidemargin}%
          \addtolength\@tempdimb{1in}%
```

```
          \setlength{\@tempdimc}{\paperwidth}%
          \addtolength{\@tempdimc}{-\textwidth}%
          \divide\@tempdimb by 32768\relax
          \divide\@tempdimc by 32768\relax
          \ifdim\@tempdimc=\z@\else
            \multiply\@tempdima by\@tempdimb
            \divide\@tempdima by\@tempdimc
          \fi
          \edef\@tempa{\noexpand\endgroup
            \noexpand\@setplength{refhpos}{\the\@tempdima}}%
        \@tempa
    \fi\fi
    \showfield{\useplength{refhpos}}{\useplength{refvpos}}%
              {\useplength{refwidth}}{-1.5\baselineskip}%
  \endgroup
}
```

`\showenvelope`
`\@showenvelope`
`\@@showenvelope`
`\@@@showenvelope`

Es kann auch ein Briefumschlag angezeigt werden. Dazu müssen verschiedene Parameter des Umschlags bekannt sein. Der Umschlag wird immer hochkant auf der aktuellen Seite angezeigt. Hierfür wird das Paket graphicx benötigt. Das Paket wird nur dann explizit geladen, wenn dies nicht bereits geschehen ist. Damit ist es möglich das Paket mit Optionen vor der LCO zu laden.

```
\@ifpackageloaded{graphicx}{}{\RequirePackage{graphicx}}
\newcommand*{\showenvelope}{}
\newcommand*{\@@showenvelope}{}
\newcommand*{\@@@showenvelope}{}
\def\showenvelope(#1,#2){%
  \@ifnextchar (%)
    {\@showenvelope({#1},{#2})}%
    {\@@@showenvelope({#1},{#2})}%
}
\def\@@@showenvelope(#1,#2){%
  \begingroup
    \def\@tempa{\endgroup}%
    \ifdim \paperwidth>#1\relax
      \LCOWarning{visualize}{%
        \string\paperwidth\space > envelope width not supported}%
    \else
      \ifdim \paperheight>#2\relax
        \ifdim .5\paperheight>#2\relax
          \expandafter\ifdim \useplength{tfoldmarkvpos}>#2\relax
            \LCOWarning{visualize}{%
              tfoldmarkvpos > envelope height not supported}%
          \else
            \def\@tempa{\endgroup
              \@showenvelope({#1},{#2})%
                              ({\dimexpr (#1-\paperwidth)/2\relax},%
                               {\dimexpr (#2-\expandafter\dimexpr
```

```
                                        \useplength{tfoldmarkvpos}\relax)%
                                       /2\relax})%
            }%
          \fi
        \else
          \def\@tempa{\endgroup
            \@showenvelope({#1},{#2})%
                          ({\dimexpr (#1-\paperwidth)/2\relax},%
                           {\dimexpr (#2-.5\paperheight)/2\relax})%
          }%
        \fi
      \else
        \def\@tempa{\endgroup
          \@showenvelope({#1},{#2})%
                        ({\dimexpr (#1-\paperwidth)/2\relax},%
                         {\dimexpr (#2-\paperheight)/2\relax})%
        }%
      \fi
    \fi
  \@tempa
}
\def\@showenvelope(#1,#2)(#3,#4){%
  \@ifnextchar [%]
    {\@@showenvelope({#1},{#2})({#3},{#4})}%
    {\@@showenvelope({#1},{#2})({#3},{#4})[]}%
}
\def\@@showenvelope(#1,#2)(#3,#4)[#5]{%
```

Die Argumente sind:

1. Breite des Umschlags,
2. Höhe des Umschlags,
3. Abstand des Briefbogens vom linken und rechten Rand des Umschlags,
4. Abstand des Briefbogens vom oberen und unteren Rand des Umschlags,
5. optionale weitere Anweisungen, zum Setzen von Elementen.

```
  \newpage
  \vspace*{\fill}
  \rotatebox{90}{%
    \begin{picture}(0,0)
      \begin{picture}(\LenToUnit{#1},\LenToUnit{#2})(0,\LenToUnit{#2})
        \newcommand*{\PlusHOffset}[1]{%
          \dimexpr \expandafter\dimexpr ##1\relax + #3\relax
        }%
        \newcommand*{\MinusHOffset}[1]{%
```

```
          \dimexpr \expandafter\dimexpr ##1\relax - #3\relax
        }%
        \newcommand*{\PlusVOffset}[1]{%
          \dimexpr \expandafter\dimexpr ##1\relax + #4\relax
        }%
        \newcommand*{\MinusVOffset}[1]{%
          \dimexpr \expandafter\dimexpr ##1\relax - #4\relax
        }%
        \newcommand{\AtEnvelopeUpperLeft}[1]{%
          \put(\LenToUnit{\PlusHOffset\z@},\LenToUnit{\MinusVOffset{#2}}){##1}%
        }%
        \newcommand{\AtEnvelopeLowerLeft}[1]{%
          \put(\LenToUnit{\PlusHOffset\z@},\LenToUnit{\PlusVOffset\z@}){##1}%
        }%
        \newcommand{\AtEnvelopeUpperRight}[1]{%
          \put(\LenToUnit{\MinusHOffset{#1}},\LenToUnit{\MinusVOffset{#2}}){##1}%
        }%
        \newcommand{\AtEnvelopeLowerRight}[1]{%
          \put(\LenToUnit{\MinusHOffset{#1}},\LenToUnit{\PlusVOffset\z@}){##1}%
        }%
        \newcommand*{\measuredIFrame}{}%
        \def\measuredIFrame(##1,##2)(##3,##4){%
          \put(\LenToUnit{##1},\LenToUnit{##2}){%
            \measuredFrameLB({##3},{##4})%
          }%
          \put(\LenToUnit{\MinusHOffset\z@},%
               \LenToUnit{\dimexpr ##2-.5mm\relax}){%
            \measureLineHB{\PlusHOffset{##1}}%
          }%
          \put(\LenToUnit{\dimexpr ##1+##3\relax},%
              \LenToUnit{\dimexpr ##2-.5mm\relax}){%
            \measureLineHB{\dimexpr #1-%
                           \PlusHOffset{\dimexpr ##1+##3\relax}\relax}%
          }%
          \put(\LenToUnit{\dimexpr ##1-.5mm\relax},%
               \LenToUnit{\dimexpr ##2+##4\relax}){%
            \measureLineVL{\PlusVOffset{\dimexpr -##2-##4\relax}}%
          }%
          \put(\LenToUnit{\dimexpr ##1-.5mm\relax},%
               \LenToUnit{\PlusVOffset{-#2}}){%
            \measureLineVL{\dimexpr #2-%
              \PlusVOffset{\dimexpr -##2\relax}\relax}%
          }%
        }%
        \put(0,0){%
          \thicklines
          \usekomafont{envelope}%
          \measuredFrameLB({#1},{#2})%
        }%
```

```
        \AtEnvelopeLowerLeft{%
          \usekomafont{letter}%
          \dashbox{\LenToUnit{1mm}}%
                  (\LenToUnit{\MinusHOffset{\MinusHOffset{#1}}},%
                   \LenToUnit{\MinusVOffset{\MinusVOffset{#2}}}){}%
        }%
        \expandafter\ifdim \useplength{toaddrhpos}<\z@
          \AtEnvelopeUpperLeft{%
            \thicklines
            \usekomafont{envelope}%
            \measuredIFrame(\expandafter\dimexpr \useplength{toaddrhpos}+
                              \MinusHOffset{\MinusHOffset{#1}}\relax,%
                            -\dimexpr %
                              \expandafter\dimexpr\useplength{toaddrvpos}\relax
                              +
                              \expandafter
                                \dimexpr\useplength{toaddrheight}\relax
                              \relax)%
                           (\useplength{toaddrwidth},\useplength{toaddrheight})%
          }%
        \else
          \AtEnvelopeUpperLeft{%
            \thicklines
            \usekomafont{envelope}%
            \measuredIFrame(\useplength{toaddrhpos},%
                            -\dimexpr %
                              \expandafter\dimexpr\useplength{toaddrvpos}\relax
                              +
                              \expandafter
                                \dimexpr\useplength{toaddrheight}\relax
                              \relax)%
                           (\useplength{toaddrwidth},\useplength{toaddrheight})%
          }%
        \fi
        \AtEnvelopeUpperLeft{#5}%
      \end{picture}
    \end{picture}
  }%
  \newpage
}
```

`envelope` Farbe des Umschlagrahmens.

```
\newkomafont{envelope}{\normalcolor}
```

`letter` Farbe des getrichelten Briefbogens im Umschlag

`\unmeasuredFrame` Ein Rahmen ohne Maßlinien.

```
\newcommand*{\unmeasuredFrame}{}
\def\unmeasuredFrame(#1,#2){%
```

```
  \put(0,0){\line(1,0){\LenToUnit{#1}}}%
  \put(\LenToUnit{#1},0){\line(0,1){\LenToUnit{#2}}}%
  \put(\LenToUnit{#1},\LenToUnit{#2}){\line(-1,0){\LenToUnit{#1}}}%
  \put(0,\LenToUnit{#2}){\line(0,-1){\LenToUnit{#2}}}%
}
```

\measuredFrameLB
\measuredFrameLT
\measuredFrameRB
\measuredFrameRT

Die Maßlinien für ein Rechteck.

```
\newcommand*{\measuredFrameLB}{}
\def\measuredFrameLB(#1,#2){%
  \unmeasuredFrame({#1},{#2})%
  \put(\LenToUnit{-.5mm},0){\measureLineVL{#2}}%
  \put(0,\LenToUnit{-.5mm}){\measureLineHB{#1}}%
}
\newcommand*{\measuredFrameLT}{}
\def\measuredFrameLT(#1,#2){%
  \unmeasuredFrame({#1},{#2})%
  \put(\LenToUnit{-.5mm},0){\measureLineVL{#2}}%
  \put(0,\LenToUnit{\expandafter\dimexpr #2+.5mm\relax}){\measureLineHT{#1}}%
}
\newcommand*{\measuredFrameRB}{}
\def\measuredFrameRB(#1,#2){%
  \unmeasuredFrame({#1},{#2})%
  \put(\LenToUnit{\expandafter\dimexpr #1+.5mm\relax},0){\measureLineVR{#2}}%
  \put(0,\LenToUnit{-.5mm}){\measureLineHB{#1}}%
}
\newcommand*{\measuredFrameRT}{}
\def\measuredFrameRT(#1,#2){%
  \unmeasuredFrame({#1},{#2})%
  \put(\LenToUnit{\expandafter\dimexpr #1+.5mm\relax},0){\measureLineVR{#2}}%
  \put(0,\LenToUnit{\expandafter\dimexpr #2+.5mm\relax}){\measureLineHT{#1}}%
}
```

\measureLineV

Vertikale Maßlinie ohne Beschriftung:

```
\newcommand*{\measureLineV}[1]{%
  \begin{picture}(0,0)
    \thinlines
    \usekomafont{measure}%
    \put(0,0){\vector(0,1){\LenToUnit{#1}}}%
    \put(0,\LenToUnit{#1}){\vector(0,-1){\LenToUnit{#1}}}%
  \end{picture}
}
```

\measureLineVL
\measureLineVR

Vertikale Maßlinie mit Beschriftung links bzw. rechts.

```
\newcommand*{\measureLineVL}[1]{%
  \begin{picture}(0,0)
    \usekomafont{measure}%
    \put(0,0){\measureLineV{#1}}%
    \put(0,0){\makebox(0,\LenToUnit{#1})[r]{\ValPerUnit{#1}}}%
  \end{picture}
```

```
}
\newcommand*{\measureLineVR}[1]{%
  \begin{picture}(0,0)
    \usekomafont{measure}%
    \put(0,0){\measureLineV{#1}}%
    \put(0,0){\makebox(0,\LenToUnit{#1})[l]{\ValPerUnit{#1}}}%
  \end{picture}
}
```

`\ValPerUnit` `\unitfactor` Zeige die Längen in `\unitlength` mit einer Genauigkeit von 1/`\unitfactor` an.

```
\newcommand*{\ValPerUnit}[1]{%
  \begingroup
    \setlength{\@tempdima}{%
      \dimexpr #1/(\unitlength/\unitfactor)*\p@/\unitfactor\relax
    }%
    \strip@pt\@tempdima
  \endgroup
}
\newcommand*{\unitfactor}{1}
```

`\measureLineH` Horizontale Maßlinie ohne Beschriftung:

```
\newcommand*{\measureLineH}[1]{%
  \begin{picture}(0,0)
    \usekomafont{measure}%
    \put(0,0){\vector(1,0){\LenToUnit{#1}}}%
    \put(\LenToUnit{#1},0){\vector(-1,0){\LenToUnit{#1}}}%
  \end{picture}
}
```

`\measureLineHB` `\measureLineHT` Horizontale Maßlinie mit Beschriftung darunter bzw. oben.

```
\newcommand*{\measureLineHB}[1]{%
  \begin{picture}(0,0)
    \usekomafont{measure}%
    \put(0,0){\measureLineH{#1}}%
    \put(0,\LenToUnit{\dimexpr -\ht\strutbox-.5mm\relax}){%
      \makebox(\LenToUnit{#1},\LenToUnit{\baselineskip})[c]{\ValPerUnit{#1}}}%
  \end{picture}
}
\newcommand*{\measureLineHT}[1]{%
  \begin{picture}(0,0)
    \usekomafont{measure}%
    \put(0,0){\measureLineH{#1}}%
    \put(0,0){%
      \makebox(\LenToUnit{#1},\LenToUnit{\baselineskip})[c]{\ValPerUnit{#1}}}%
  \end{picture}
}
```

`measure` Farbe der Maßlinien.

```
\newkomafont{measure}{\normalcolor}
```

\showISOenvelope Zeigt einen DIN/ISO C4, C5, DL=C5/6, C6/5 Umschlag.

```
\newcommand*{\showISOenvelope}[1]{%
  \ifstr{#1}{C4}{%
    \showenvelope(324mm,229mm)%
  }{%
    \ifstr{#1}{C5}{%
      \showenvelope(229mm,162mm)%
    }{%
      \ifstr{#1}{C5/6}{%
        \showenvelope(220mm,110mm)%
      }{%
        \ifstr{#1}{DL}{%
          \showenvelope(220mm,110mm)%
        }{%
          \ifstr{#1}{C6/5}{%
            \showenvelope(229mm,114mm)%
          }{%
            \ifstr{#1}{C6}{%
              \showenvelope(162mm,114mm)%
            }{%
              \LCOWarning{visualize}{envelope size ‘ISO #1’ unsupported}%
            }%
          }%
        }%
      }%
    }%
  }%
}
```

\showUScommercial Zeige ein US commercial Umschlag der Nummer 9 und 10

```
\newcommand*{\showUScommercial}[1]{%
  \ifstr{#1}{9}{%
    \showenvelope(8.875in,3.875in)%
  }{%
    \ifstr{#1}{10}{%
      \showenvelope(9.5in,4.125in)%
    }{%
      \LCOWarning{visualize}{envelope size ‘US commercial #1’ unsupported}%
    }%
  }%
}
```

\showUScheck Zeige einen US check Umschag.

```
\newcommand*{\showUScheck}{%
  \showenvelope(8.625in,3.625in)%
}
```

`\showUSletterCixDW` Ein Umschlag im Format commercial No. 9 mit 2 Fenstern, Absenderfenster $3\frac{1}{2}$ in $\times$ $\frac{7}{8}$ in, $\frac{5}{16}$ in von links und $2\frac{1}{2}$ in von unten.

```
⟨*UScommercial9DW⟩
\newcommand*{\showUSletterCixDW}{%
  \showUScommercial9[{%
    \thicklines
    \usekomafont{envelope}%
    \measuredIFrame({\dimexpr (\paperwidth-
                          \expandafter\dimexpr \useplength{firstheadwidth}\relax%
                        )/2\relax},%
                     -\expandafter\dimexpr\useplength{firstheadvpos}\relax)%
                    (3.5in,.875in)%
  }]%
}
⟨/UScommercial9DW⟩
```

lco file	Envelope			Window			
	Name	Length	Width	H-pos	V-pos	Width	Height
NipponEL	Chou/You 3	235 mm	120 mm	22 mm	12 mm	90 mm	45 mm
NipponEH	Chou/You 3	235 mm	120 mm	22 mm	12 mm	90 mm	55 mm
NipponLL	Chou/You 3	235 mm	120 mm	25 mm	12 mm	90 mm	45 mm
NipponLH	Chou/You 3	235 mm	120 mm	25 mm	12 mm	90 mm	55 mm
NipponEL	Chou/You 4	205 mm	90 mm	22 mm	12 mm	90 mm	45 mm
NipponEH	Chou/You 4	205 mm	90 mm	22 mm	12 mm	90 mm	55 mm
NipponLL	Chou/You 4	205 mm	90 mm	25 mm	12 mm	90 mm	45 mm
NipponLH	Chou/You 4	205 mm	90 mm	25 mm	12 mm	90 mm	55 mm
NipponRL	Chou/You 4	205 mm	90 mm	98 mm	28 mm	90 mm	45 mm
KakuLL	Kaku-kata	312 mm	228 mm	25 mm	24 mm	90 mm	45 mm

Tabelle 2: The supported Japanese envelopes

Datei x
japanlco.dtx

24 Option Files for Japanese Letters

Gernot Hassenpflug wrote a lot of lco files for use of Japanese envelope formats. Markus Kohm, the author of KOMA-Script, addapted them to an extended version of scrlttr2 and wrote a dtx file with the moste recent lco files. Table 2 shows all lco files of the supported envelopes with their datas.

24.0.1 The File Header

Declaration of the file.

```
\ProvidesFile{%
⟨Ni⟩ Nippon%
⟨Kaku⟩ Kaku%
⟨EARLY⟩ E%
⟨LATE⟩ L%
⟨RIGHT⟩ R%
⟨LOW⟩ L%
⟨HIGH⟩ H%
}[\KOMAScriptVersion\space Japanese letter-class-option]
```

\LCOWarningNoLine
\LCOWarning

We need macros that are similar to \ClassWarningNoLine or \PackageWarningNoLine and \ClassWarning or \PackageWarning

24.0.2 Main-Part

Check the class.

```
\@ifundefined{scr@fromname@var}{%
  \LCOWarningNoLine{%
⟨Ni⟩  Nippon%
⟨Kaku⟩  Kaku%
⟨EARLY⟩  E%
⟨LATE⟩  L%
⟨RIGHT⟩  R%
⟨LOW⟩  L%
⟨HIGH⟩  H%
  }{%
    This letter class option file was made only\MessageBreak
    to be used with KOMA-Script letter class\MessageBreak
    ‘scrlttr2’. Use with other classes may result\MessageBreak
    in a lot of errors%
  }%
}{}
```

Is the expected paper size used?

```
\LetterOptionNeedsPapersize{%
⟨Ni⟩  Nippon%
⟨Kaku⟩  Kaku%
⟨EARLY⟩  E%
⟨LATE⟩  L%
⟨RIGHT⟩  R%
⟨LOW⟩  L%
⟨HIGH⟩  H%
}{a4}
```

All the values

```
\setkomavar*{fromzipcode}{J}
```

```
\@setplength{foldmarkhpos}{3.5mm}
\@setplength{tfoldmarkvpos}{83mm}
\@setplength{tfoldmarklength}{4mm}
\@setplength{mfoldmarkvpos}{116mm}
\@setplength{mfoldmarklength}{2mm}
\@setplength{bfoldmarkvpos}{232mm}
\@setplength{bfoldmarklength}{2mm}
\@setplength{foldmarkvpos}{3.5mm}
\@setplength{lfoldmarkhpos}{202mm}
⟨!Ni|!RIGHT⟩\@setplength{refvpos}{69mm}
⟨Ni & RIGHT⟩\@setplength{refvpos}{84mm}
\@setplength{refaftervskip}{\baselineskip}
⟨EARLY⟩\@setplength{toaddrhpos}{22mm}
⟨LATE⟩\@setplength{toaddrhpos}{25mm}
⟨RIGHT⟩\@setplength{toaddrhpos}{-22mm}
\@setplength{toaddrwidth}{90mm}
⟨Ni&!RIGHT⟩\@setplength{toaddrvpos}{12mm}
```

```
⟨Ni & RIGHT⟩\@setplength{toaddrvpos}{28mm}
⟨Kaku⟩\@setplength{toaddrvpos}{24mm}
⟨LOW⟩\@setplength{toaddrheight}{40mm}
⟨HIGH⟩\@setplength{toaddrheight}{50mm}
\@setplength{toaddrindent}{\z@}
\@setplength{backaddrheight}{5mm}
\@setplength{specialmailindent}{\fill}
\@setplength{specialmailrightindent}{1em}
\@setplength{locwidth}{\z@}
⟨Ni⟩\@setplength{locheight}{55mm}%
⟨Kaku⟩\@setplength{locheight}{45mm}%
\@setplength{firstheadvpos}{8mm}
\@setplength{firstheadwidth}{\paperwidth}
\ifdim \useplength{toaddrhpos}>\z@
  \@addtoplength[-2]{firstheadwidth}{\useplength{toaddrhpos}}
\else
  \@addtoplength[2]{firstheadwidth}{\useplength{toaddrhpos}}
\fi
\@setplength{firstfootwidth}{\useplength{firstheadwidth}}
\@setplength{firstfootvpos}{\paperheight}
\@addtoplength{firstfootvpos}{-16mm}
\@setplength{refwidth}{\z@}
\@setplength{sigindent}{\z@}
\@setplength{sigbeforevskip}{2\baselineskip}
\let\raggedsignature=\centering
```

Additional settings:

```
\ifcase\@alignfrom
  \KOMAoptions{fromalign=locationright}%
\or
  \KOMAoptions{fromalign=locationright}%
\or
  \KOMAoptions{fromalign=locationright}%
\fi
```

Datei y scrkcile.dtx

25 Serienbriefe und Adressdateien

Die KOMA-Script-Brief-Klasse unterstützt die Verwendung von Adressdateien beispielsweise für Serienbriefe.

25.1 Option

Die Implementierung ist von Optionen unabhängig.

25.2 Makos für Serienbriefe und Adressdateien

`\adrentry` `\adrchar`

Serienbriefe werden mit Hilfe der Funktionen `\adrentry` und `\adrchar` und einer Adressdatei realisiert. Darüber hinaus werden mit diesen Befehlen Abkürzungen für Adressen definiert. Durch die Definition mit Hilfe von `\addrentry` bzw. `\addrchar` müssen vom Anwender für neue Anwendungen nur die neuen Befehle umdefiniert werden.

```
\newcommand*{\adrentry}[7]{%
  \addrentry{#1}{#2}{#3}{#4}{#5}{#6}{#7}{}}
\newcommand*{\adrchar}{\addrchar}
```

`\addrentry` `\addrchar`

Diese Makros ermöglichen ab Version 2.8n die Verwendung von bis zu 9 Argumenten.

```
\newcommand*{\addrentry}[9]{\def\@tempa{#1}\ifx \@tempa\@empty \else
 \def\@tempa{#2}\ifx \@tempa\@empty
  \expandafter\def\csname #9\endcsname{#1\\#3}%
 \else
  \expandafter\def\csname #9\endcsname{#2 #1\\#3}%
 \fi \fi}
\newcommand*{\addrchar}[1]{}
```

Datei z scrextend.dtx

26 Erweiterungen für andere Klassen

Dieses Paket dient dazu, einige Möglichkeiten von KOMA-Script auch anderen Klassen zur Verfügung zustellen. Dafür darf dieses Paket nicht mit einer KOMA-Script-Klasse verwendet werden.

26.1 Vorbereitung noch vor der Identifizierung des Pakets

Die Vorbereitung ist einfach. Es wird nur getestet, ob bereits eine KOMA-Script-Klasse geladen ist. Falls dem so ist, wird eine Warnung ausgegeben und dafür gesorgt, dass das Paket nicht weiter abgearbeitet wird.

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname KOMAClassName\endcsname\relax\else
  \PackageWarning{scrextend}{%
    This package was made to extend other classes but\MessageBreak
    KOMA-Script classes by some features of the\MessageBreak
    KOMA-Script classes.\MessageBreak
    It should never be used with a KOMA-Script class.\MessageBreak
    Loading of the package will be aborted\@gobble%
  }%
  \expandafter\endinput
\fi
```

26.2 Identifizierung des Pakets

```
\ProvidesPackage{scrextend}[\KOMAScriptVersion\space KOMA-Script package
  (extend other classes with features of KOMA-Script classes)]
```

26.3 Vorbereitung vor den Optionen

\scr@ext@activate
\scr@ext@activateable
\scr@ext@addto@activateable

Einige Dinge werden nur dann definiert, wenn dies gefordert ist. Dazu werden die entsprechenden Definitionen in das zweite Argument von `\scr@ext@activateable{⟨Kürzel⟩}` gepackt, wodurch sie als `\scr@ext@activate@⟨Kürzel⟩` definiert werden, und anschließend `\csname scr@ext@immediate@⟨Kürzel⟩\endcsname` ausgeführt. Um das dann zu aktivieren wird `\scr@ext@activate{⟨Kürzel⟩}` aufgerufen (beispielsweise innerhalb einer Option). Mit `\scr@ext@addto@activateable` können Aktivierungen entsprechend um weitere Definitionen erweitert werden.

```
\newcommand*{\scr@ext@activate}[1]{%
  \scr@ifundefinedorrelax{scr@ext@activate@#1}{%
    \@namedef{scr@ext@immediate@#1}{%
```

```
      \expandafter\let\csname scr@ext@immediate@#1\endcsname\relax
      \csname scr@ext@activate@#1\endcsname
    }%
  }{%
    \csname scr@ext@activate@#1\endcsname
    \expandafter\let\csname scr@ext@activate@#1\endcsname\relax
  }%
}
\newcommand*{\scr@ext@activateable}[1]{%
  \expandafter\newcommand\expandafter*\csname scr@ext@activate@#1\endcsname
}
\newcommand*{\scr@ext@addto@activateable}[1]{%
  \scr@ifundefinedorrelax{scr@ext@activate@#1}{%
    \expandafter\newcommand\expandafter*\csname scr@ext@activate@#1\endcsname
  }{%
    \expandafter\g@addto@macro\csname scr@ext@activate@#1\endcsname
  }%
}
```

26.4 Optionen

Das Paket kennt eine universelle Option, um aktivierbare Möglichkeiten zu aktivieren. Diese Option steht aber nur während dem Laden des Pakets zur Verfügung!

`extendedfeature`

```
\KOMA@key{extendedfeature}{%
  \scr@ext@activate{#1}%
  \AtEndOfPackage{%
    \scr@ifundefinedorrelax{scr@ext@immediate@#1}{}{%
      \PackageErrorNoLine{unkown extended feature ‘#1’}{%
        You have used option ‘extendedfeature’ to activate an extended feature
        named\MessageBreak
        ‘#1’, but there is no such extended feature.\MessageBreak
        See the KOMA-Script manual for more infromation about known
        activatable\MessageBreak
        extended features%
      }%
    }%
  }%
}
\AtEndOfPackage{%
  \KOMA@key{extendedfeature}{%
    \PackageError{scrextend}{%
      Option ‘extendedfeature’ too late%
    }{%
      Option ‘extendedfeature’ my be used only while loading the
      package.\MessageBreak
      But you’ve tried to set it up later%
```

```
    }%
  }%
}
```

Datei A
scrhack.dtx

KOMA-Script File scrsource *

Markus Kohm†

2011-02-19

Some packages from other authors may have problems with KOMA-Script. In my opinion some packages could be improved. With some packages this makes only sense, if KOMA-Script was used. With some other packages the package author has another opinion. Sometimes proposals was never answered. Package scrhack contains all those improvement proposals for other packages. This means, scrhack redefines macros of packages from other authors! The redefinitions are only activated, if those packages were loaded. Users may prevent scrhack from redefining macros of individual packages.

Contents

27 The hyperref hack

Before version 6.79h package hyperref does behave different at part, chapter, and section headings that get no number. If they get no number, because of to low counter `secnumdepth` hyperref sets an anchor for links and bookmarks before the heading. Same would be, if the headings have a number. But if the headings get no number because of usage of the star version of the commands, e.g., `\part*`, `\chapter*` or `\section*`, the anchor for links and bookmarks are set after the headings. The anchors for numbered headings are always set before the headings.

Package scrhack redefines some macros of some hyperref driver files, e.g., hpdftex.def, after loading the hyperref driver file. With this redefinitions the anchor of not numbered headings will be set always before the headings, too.

*This file is version 3.08a of `scrsource`.

†mailto:komascript(at)gmx.info

You may switch off the hyperref hack loading package scrhack with option `hyperref=false`. You may also switch off the hyperref hack using `\KOMAoptions{hyperref=false}` or `\KOMAoption{hyperref}{false}` somewhere after loading package scrhack, but before loading the hyperref driver package, that is by default after loading the package.

28 The float hack

Package float uses macros `\float@listhead` to set the headings of a float listing and `\float@addtolists` to add informations to all float listings. These macros where proposed by the KOMA-Script author for some years. In theory those macros may be used by several class and package authors to deligate some parts of the creation of a float listing to the class. This would increase the compatiblity of packages and classes. But unfortunately some package authors, even the author of package float, implemented the commands in such a way, that these packages will become incompatible to each other.

Because of this KOMA-Script stopped support for `\float@addtolists` and `\float@listhead` with version 3. Instead of this KOMA-Script supports several improvements for package authors using KOMA-Script package tocbasic.

Package scrhack redefines some macros of package float to not longer use `\float@addtolists` and `\float@listhead` but use the interface of package tocbasic. This does not only improve the compatibility of KOMA-Script and package float, but also improves the compatibility of packages babel and float.

Note: A significant change with scrhack is, that KOMA-Script options like `lists=totoc` or `lists=totocnumbered` does only change the behaviour of float listings, that are already defined using `\newfloat` *before* using such an option!

You may switch off the float hack loading package scrhack with option `float=false`. You may also switch off the float hack using `\KOMAoptions{float=false}` or `\KOMAoption{float}{false}` somewhere after loading package scrhack, but before loading package float.

29 The listings hack

Package listings uses macros `\float@listhead` to set the headings of a float listing, if defined, and `\float@addtolists` to add informations to all float listings. These macros where proposed by the KOMA-Script author for some years. In theory those macros may be used by several class and package authors to deligate some parts of the creation of a float listing to the class. This would increase the compatiblity of packages and classes.

But unfortunately some package authors, even the author of package float, impemented the commands in such a way, that these packages may become incompatible to each other.

Because of this KOMA-Script stopped support for `\float@addtolists` and `\float@listhead` with version 3. Instead of this KOMA-Script supports several improvements for package authors using KOMA-Script package tocbasic.

Package scrhack redefines some macros of package listings to not longer use `\float@addtolists` and `\float@listhead` but use the interface of package tocbasic. This does not only improve the compatibility of KOMA-Script and package listings, but also improves the compatibility of packages babel and listings.

Note: A significant change with scrhack is, that KOMA-Script options like `lists=totoc` or `lists=totocnumbered` does only change the behaviour of `\listoflistings`, if they are set after loading package listings.

You may switch off the listings hack loading package scrhack with option `listings=false`. You may also switch off the listings hack using `\KOMAoptions{listings=false}` or `\KOMAoption{listings}{false}` somewhere after loading package scrhack, but before loading package listings.

30 Implementation of scrhack

30.1 Optionen

Das Paket bedient sich `\KOMAoptions` etc. aus scrkbase (dieses wird übrigens direkt per `scrkbase.dtx` geladen).

Per Option kann gewählt werden, welche Manipulationen geladen werden sollen. Alle diese Optionen können jedoch nur bis zum Laden des entsprechenden Pakets oder dem Laden von scrhack gesetzt werden (es zählt, was später kommt). Anschließend sind sie wirkungslos.

30.2 Verwendete Anweisungen

`\scr@ifexpected` Wenn die im ersten Argument angegebene Anweisung nach Ausführung der im zweiten Argument angegebenen Anweisungen unverändert ist, dann soll das dritte Argument ausgeführt werden, sonst das vierte.

```
\newcommand{\scr@ifexpected}[2]{%
  \begingroup
    \let\@tempa#1
    #2
    \ifx\@tempa#1
      \aftergroup\@firstoftwo
    \else
      \aftergroup\@secondoftwo
```

```
    \fi
  \endgroup
}
```

\scr@hack@load

Wenn die Datei mit dem Namen des zweiten Arguments und der Endung des ersten Arguments so geladen wurde, dass LaTeX eine Versionsinfo dazu gespeichert hat, dann soll zusätzlich der entsprechende Hack geladen werden.

```
\newcommand*{\scr@hack@load}[2]{%
  \expandafter\ifx\csname ver@#2.#1\endcsname\relax
    \expandafter\@secondoftwo
  \else
    \expandafter\@firstoftwo
  \fi
  {%
    \PackageInfo{scrhack}{loading #2 hack}%
    \edef\reserved@a{%
      \noexpand\makeatletter\noexpand\input{#2.hak}%
      \noexpand\catcode`\noexpand\@\the\catcode`\@\relax
    }\reserved@a
  }{%
    \PackageInfo{scrhack}{ignorring #2 hack}%
  }%
}
```

30.3 Der hyperref-Hack

hyperref setzt den Anker zu der Stern-Variante einer Überschrift hinter die Überschrift, während es bei der nicht Stern-Variante den Anker auch dann vor die Überschrift setzt, wenn die Überschrift aufgrund von `secnumdepth` nicht nummeriert wird. Der Hack setzt den Anker einheitlich vor die Überschrift.

hyperref

```
⟨*package & option⟩
\KOMA@ifkey{hyperref}{@scrhack@hyperref}%
\@scrhack@hyperreftrue
⟨/package & option⟩
⟨*package & body⟩
```

Hier muss ein wenig trickreicher gearbeitet werden, weil hyperref die Treiberdatei per `\AtEndOfPackage` lädt und der Hack erst danach installiert werden darf. Mit `\AfterPackage*` alleine, würde der Hack aber vor dem Laden der Treiberdatei installiert. Dafür können wir aber sicher sein, dass ein innerhalb von `\AfterPackage*` aufgerufenes `\AtEndOfPackage` garantiert nach dem Laden der Treiberdatei ausgeführt wird. Das funktioniert auch noch, wenn hyperref bereits geladen wurde. In dem Fall wird der Co-

de einfach nach dem Ende von scrhack statt nach dem Ende von hyperref ausgeführt.

```
\AfterPackage*{hyperref}{%
  \@ifpackagelater{hyperref}{2009/11/24}{%
    \PackageInfo{scrhack}{hyperref hack deactivated because of\MessageBreak
      detection of hyperref version, that doesn't\MessageBreak
      need that hack,}%
  }{%
    \AtEndOfPackage{%
      \KOMA@key[.scrhack.sty]{hyperref}{%
        \PackageWarning{scrhack}{option `hyperref=#1' ignored}%
      }%
      \if@scrhack@hyperref\scr@hack@load\@pkgextension{hyperref}\fi
    }%
  }%
}
⟨/package & body⟩
```

\@schapter
\@spart
\@ssect

Eigentlich wird hier gar nicht `hyperref.sty` verändert, sondern diverse Treiberdateien. Sobald das Paket hyperref geladen ist, ist auch die passende Treiberdatei geladen und außerdem sind alle Treiberdateien, die entsprechende Definitionen vornehmen, gleichermaßen betroffen. Also kann der entsprechende Patch einfach erfolgen, wenn hyperref geladen ist (was bereits von `\scr@hack@load` getestet wurde). Es muss also nur noch sichergestellt werden, dass die umzudefinierenden Macros derzeit den erwarteten Inhalt haben.

```
⟨*hyperref & body⟩
\scr@ifexpected\@schapter{%
  \def\@schapter#1{%
    \H@old@schapter{#1}%
    \begingroup
      \let\@mkboth\@gobbletwo
      \Hy@GlobalStepCount\Hy@linkcounter
      \xdef\@currentHref{\Hy@chapapp*.\the\Hy@linkcounter}%
      \Hy@raisedlink{%
        \hyper@anchorstart{\@currentHref}\hyper@anchorend
      }%
    \endgroup
  }%
}{%
  \PackageInfo{scrhack}{redefining \string\@schapter}%
  \def\@schapter#1{%
    \begingroup
      \let\@mkboth\@gobbletwo
      \Hy@GlobalStepCount\Hy@linkcounter
      \xdef\@currentHref{\Hy@chapapp*.\the\Hy@linkcounter}%
      \Hy@raisedlink{%
        \hyper@anchorstart{\@currentHref}\hyper@anchorend
```

```
      }%
    \endgroup
    \H@old@schapter{#1}%
  }%
}{%
  \scr@ifexpected\@schapter{%
    \def\@schapter#1{%
      \begingroup
        \let\@mkboth\@gobbletwo
        \Hy@GlobalStepCount\Hy@linkcounter
        \xdef\@currentHref{\Hy@chapapp*.\the\Hy@linkcounter}%
        \Hy@raisedlink{%
          \hyper@anchorstart{\@currentHref}\hyper@anchorend
        }%
      \endgroup
      \H@old@schapter{#1}%
    }%
  }{}{%
    \PackageWarningNoLine{scrhack}{unknown \string\@schapter\space
      definition found!\MessageBreak
      Maybe you are using a unsupported hyperref version}%
  }%
}

\scr@ifexpected\@spart{%
  \def\@spart#1{%
    \H@old@spart{#1}%
    \Hy@GlobalStepCount\Hy@linkcounter
    \xdef\@currentHref{part*.\the\Hy@linkcounter}%
    \Hy@raisedlink{%
      \hyper@anchorstart{\@currentHref}\hyper@anchorend
    }%
  }%
}{%
  \PackageInfo{scrhack}{redefining \string\@spart}%
  \def\@spart#1{%
    \Hy@GlobalStepCount\Hy@linkcounter
    \xdef\@currentHref{part*.\the\Hy@linkcounter}%
    \Hy@raisedlink{%
      \hyper@anchorstart{\@currentHref}\hyper@anchorend
    }%
    \H@old@spart{#1}%
  }%
}{%
  \scr@ifexpected\@spart{%
    \def\@spart#1{%
      \Hy@GlobalStepCount\Hy@linkcounter
      \xdef\@currentHref{part*.\the\Hy@linkcounter}%
      \Hy@raisedlink{%
```

```
        \hyper@anchorstart{\@currentHref}\hyper@anchorend
      }%
      \H@old@spart{#1}%
    }%
  }{}{%
    \PackageWarningNoLine{scrhack}{unknown \string\@spart\space
      definition found!\MessageBreak
      Maybe you are using a unsupported hyperref version}%
  }%
}

\scr@ifexpected\@ssect{%
  \def\@ssect#1#2#3#4#5{%
    \H@old@ssect{#1}{#2}{#3}{#4}{#5}%
    \phantomsection
  }%
}{%
  \PackageInfo{scrhack}{redefining \string\@ssect}%
  \def\@ssect#1#2#3#4#5{%
    \H@old@ssect{#1}{#2}{#3}{#4}{\phantomsection\ignorespaces#5}%
  }%
}{%
  \scr@ifexpected\@ssect{%
    \def\@ssect#1#2#3#4#5{%
      \H@old@ssect{#1}{#2}{#3}{#4}{\phantomsection\ignorespaces#5}%
    }%
  }{}{%
    \PackageWarningNoLine{scrhack}{unknown \string\@ssect\space
      definition found!\MessageBreak
      Maybe you are using a unsupported hyperref version}%
  }%
}
⟨/hyperref & body⟩
```

30.4 Der float-Hack

Das float-Paket verwendet das Makro `\float@listhead` zum Setzen der Überschriften. Dies wird seit KOMA-Script 3 nicht mehr empfohlen und fliegt demnächst komplett aus der Unterstützung. Stattdessen wird empfohlen, dass Pakete tocbasic unterstützen. Der Aufwand dafür ist sehr gering und wird mit vielen neuen Möglichkeiten belohnt.

Dieser Hack rüstet die tocbasic-Unterstützung für float nach.

`float`

```
⟨*package & option⟩
\KOMA@ifkey{float}{@scrhack@float}%
\@scrhack@floattrue
⟨/package & option⟩
```

```
⟨*package & body⟩
\AfterPackage*{float}{%
  \KOMA@key[.scrhack.sty]{float}{%
    \PackageWarning{scrhack}{option ‘float’ ignored}%
  }%
  \if@scrhack@float\scr@hack@load\@pkgextension{float}\fi
}
⟨/package & body⟩
```

`\newfloat` Über die Anweisung `\newfloat` wird eine neue Gleitumgebung definiert. Hier muss die neue Erweiterung aus dem dritten Argument tocbasic bekannt gemacht werden.

`\listof` Über die Anweisung `\listof` wird ein Verzeichnis für Gleitumgebungen ausgegeben. Hier muss schlicht die entsprechende Anweisung von tocbasic verwendet werden.

`\float@addtolists` Diese Anweisung wird nicht länger benötigt und daher auf die ursprüngliche Definition zurückgesetzt.

```
⟨*float & body⟩
\scr@ifexpected{\newfloat}{%
  \long\def\newfloat#1#2#3{\@namedef{ext@#1}{#3}
    \let\float@do=\relax
    \xdef\@tempa{\noexpand\float@exts{\the\float@exts \float@do{#3}}}%
    \@tempa
    \floatplacement{#1}{#2}%
    \@ifundefined{fname@#1}{\floatname{#1}{#1}}{}
    \expandafter\edef\csname ftype@#1\endcsname{\value{float@type}}%
    \addtocounter{float@type}{\value{float@type}}
    \restylefloat{#1}%
    \expandafter\edef\csname fnum@#1\endcsname%
    {\expandafter\noexpand\csname fname@#1\endcsname{}
      \expandafter\noexpand\csname the#1\endcsname}
    \@ifnextchar[%]
    {\float@newx{#1}}%
    {\@ifundefined{c@#1}{\newcounter{#1}\@namedef{the#1}{\arabic{#1}}}%
      {}}}%
}{%
  \scr@ifexpected{\listof}{%
    \def\listof#1#2{%
      \@ifundefined{ext@#1}{\float@error{#1}}{%
        \@namedef{l@#1}{\@dottedtocline{1}{1.5em}{2.3em}}%
        \float@listhead{#2}%
        \begingroup\setlength{\parskip}{\z@}%
        \@starttoc{\@nameuse{ext@#1}}%
        \endgroup}}%
  }{%
    \RequirePackage{tocbasic}%
    \PackageInfo{scrhack}{redefining \string\newfloat}%
```

```
  \renewcommand\newfloat[3]{%
    \ifattoclist{#3}{%
      \PackageError{scrhack}{extension `#3' already in use}{%
        Each extension may be used only once.\MessageBreak
        You, the class, or another package already uses extension
        `#3'.\MessageBreak
        \string\newfloat\space command will be ignored!}%
    }{%
      \addtotoclist[float]{#3}%
      \setuptoc{#3}{chapteratlist}%
      \@namedef{ext@#1}{#3}%
      \let\float@do=\relax
      \xdef\@tempa{\noexpand\float@exts{\the\float@exts \float@do{#3}}}%
      \@tempa
      \floatplacement{#1}{#2}%
      \@ifundefined{fname@#1}{\floatname{#1}{#1}}{}%
      \expandafter\edef\csname ftype@#1\endcsname{\value{float@type}}%
      \addtocounter{float@type}{\value{float@type}}
      \restylefloat{#1}%
      \expandafter\edef\csname fnum@#1\endcsname%
      {\expandafter\noexpand\csname fname@#1\endcsname{}
        \expandafter\noexpand\csname the#1\endcsname}%
      \@ifnextchar[%]
      {\float@newx{#1}}%
      {\@ifundefined{c@#1}{\newcounter{#1}\@namedef{the#1}{\arabic{#1}}}%
        {}}}%
  }%
  \PackageInfo{scrhack}{redefining \string\listof}%
  \renewcommand*\listof[2]{%
    \@ifundefined{ext@#1}{\float@error{#1}}{%
      \@ifundefined{l@#1}{\expandafter\let\csname l@#1\endcsname\l@figure
        \@ifundefined{l@#1}{%
          \@namedef{l@#1}{\@dottedtocline{1}{1.5em}{2.3em}}}{}%
      }{}%
      \listoftoc[{#2}]{\csname ext@#1\endcsname}%
    }%
  }%
  \scr@ifexpected{\float@addtolists}{%
    \long\def\float@addtolists#1{%
      \def\float@do##1{\addtocontents{##1}{#1}} \the\float@exts}%
  }{%
    \PackageInfo{scrhack}{undefining \string\float@addtolists}%
    \let\float@addtolists\relax
  }{%
    \PackageWarningNoLine{scrhack}{unkown \string\float@addtolists\space
      definition found!\MessageBreak
      Maybe you are using a unsupported float version}%
  }%
}{%
```

```
    \PackageWarningNoLine{scrhack}{unknown \string\listof\space
      definition found!\MessageBreak
      Maybe you are using a unsupported float version}%
  }%
}{%
  \PackageWarningNoLine{scrhack}{unknown \string\newfloat\space
    definition found!\MessageBreak
    Maybe you are using a unsupported float version}%
}
⟨/float & body⟩
```

30.5 Der listings-Hack

Das listings-Paket verwendet das Makro `\float@listhead` zum Setzen der Überschriften. Dies wird seit KOMA-Script 3 nicht mehr empfohlen und fliegt demnächst komplett aus der Unterstützung. Stattdessen wird empfohlen, dass Pakete tocbasic unterstützen. Der Aufwand dafür ist sehr gering und wird mit vielen neuen Möglichkeiten belohnt.

Dieser Hack rüstet die tocbasic-Unterstützung für listings nach.

`listings`

```
⟨*package & option⟩
\KOMA@ifkey{listings}{@scrhack@listings}%
\@scrhack@listingstrue
⟨/package & option⟩
⟨*package & body⟩
\AfterPackage*{listings}{%
  \KOMA@key[.scrhack.sty]{listings}{%
    \PackageWarning{scrhack}{option ‘listings’ ignored}%
  }%
  \if@scrhack@listings\scr@hack@load\@pkgextension{listings}\fi
}
⟨/package & body⟩
```

`\scr@do@hack@listings`
`\lstlistoflistings`

Über dieses Macro wird das Verzeichnis der Listings gesetzt. Die gesamte Funktionalität dafür kann tocbasic überlassen werden.

`\float@addtolists`

Diese Anweisung wird nicht länger benötigt und daher auf die ursprüngliche Definition zurückgesetzt. Da listings ihre Definition mit `\AtBeginDocument` verzögert, muss dies hier ebenfalls geschehen.

```
⟨*listings & body⟩
\newcommand*{\scr@do@hack@listings}{%
  \RequirePackage{tocbasic}%
  \addtotoclist[float]{lol}%
  \setuptoc{lol}{chapteratlist}%
  \PackageInfo{scrhack}{redefining \string\lstlistoflistings}%
  \renewcommand*{\lstlistoflistings}{\listoftoc[{\lstlistlistingname}]{lol}}%
  \AtBeginDocument{%
```

```
    \scr@ifexpected{\float@addtolists}{%
      \def\float@addtolists##1{\addtocontents{lol}{##1}}%
    }{%
      \PackageInfo{scrhack}{undefining \string\float@addtolists}%
      \let\float@addtolists\relax
    }{%
      \scr@ifexpected{\float@addtolists}{%
        \def\float@addtolists##1{\addtocontents{lol}{##1}%
          \orig@float@addtolists{##1}}%
      }{%
        \PackageInfo{scrhack}{setting \string\float@addtolists\MessageBreak
          to \string\orig@float@addtolists}%
        \let\float@addtolists\orig@float@addtolists
      }{%
        \PackageWarningNoLine{scrhack}{unkown \string\float@addtolists\space
          definition found!\MessageBreak
          Maybe you are using a unsupported listings version}%
      }%
    }%
  }%
  \let\scr@do@hack@listings\relax
}
\scr@ifexpected{\lstlistoflistings}{%
  \def\lstlistoflistings{\bgroup
    \let\contentsname\lstlistlistingname
    \let\lst@temp\@starttoc \def\@starttoc##1{\lst@temp{lol}}%
    \tableofcontents \egroup}%
}{%
  \scr@do@hack@listings
}{%
  \scr@ifexpected{\lstlistoflistings}{%
    \def\lstlistoflistings{%
      \begingroup
        \@ifundefined{@restonecoltrue}{}{%
          \if@twocolumn
            \@restonecoltrue\onecolumn
          \else
            \@restonecolfalse
          \fi
        }%
        \float@listhead{\lstlistlistingname}%
        \parskip\z@\parindent\z@\parfillskip \z@ \@plus 1fil%
        \@starttoc{lol}%
        \@ifundefined{@restonecoltrue}{}{%
          \if@restonecol\twocolumn\fi
        }%
      \endgroup
    }%
  }{%
```

```
    \scr@do@hack@listings
  }{%
    \PackageWarningNoLine{scrhack}{unknown \string\lstlistoflistings\space
      definition found!\MessageBreak
      Maybe you are using a unsupported listings version}%
  }%
}
```
⟨/listings & body⟩

30.6 Optionen ausführen

Zum Schluss noch die Optionen ausführen. Im Paket wird diese Anweisung allerdings vor den Anweisungen der Hacks und den Anweisungen aus dem Abschnitt »Verwendete Anweisungen« stehen.

⟨*package & option⟩
```
\KOMAProcessOptions\relax
```
⟨/package & option⟩

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols

\@@@@captionbeside g-70
\@@@captionbeside g-70
\@@@showenvelope w-575
\@@BCOR f-14
\@@DIV f-24
\@@PapersizeWarning w-51
\@@captionbeside g-70
\@@headlines f-34
\@@makecaption g-233
\@@makechapterhead l-467
\@@makefnmark h-123
\@@makeschapterhead l-641
\@@maybeautodot l-239
\@@setcapmargin g-316
\@@setcaptindent g-178
\@@setkomavar u-35
\@@showenvelope w-575
\@@xtrimtempa t-1445
\@AfterClass q-78
\@AfterPackage q-78
\@BCOR f-14
\@CheckKOMAScriptVersion ... a-1
\@DIV f-24
\@PapersizeWarning w-51
\@abstrtfalse k-26
\@abstrttrue k-26
\@addchap l-665
\@addmargin p-102
\@addpart l-437
\@addrfield t-774
\@addsec l-717
\@addtoplength v-43
\@afieldfalse t-24
\@afieldmode t-24
\@afieldtrue t-24
\@alignfrom t-2
\@altsecnumformatfalse l-231
\@altsecnumformattrue l-231
\@atdocumentfalse b-207
\@atdocumenttrue b-207
\@autodotfalse l-231
\@autodottrue l-231
\@baddrfalse t-78
\@baddrtrue t-78
\@beginletterhook t-1411
\@bib@break n-190
\@bigheadings l-201
\@biglocfalse t-107
\@bigloctrue t-107
\@captionbeside g-70
\@chapapp l-336
\@chapter l-467
\@chapterlistsgap m-148
\@chapterprefixfalse l-64
\@chapterprefixtrue l-64
\@datefield t-1154
\@dblfpbot(Länge) e-185
\@dblfpsep(Länge) e-185
\@dblfptop(Länge) e-185
\@dedication k-30
\@dotsep m-176
\@dynlistfalse m-44
\@dynlisttrue m-44
\@emailfalse t-21
\@emailtrue t-21
\@endpart l-337
\@enlargefpfalse t-245
\@enlargefptrue t-245
\@extratitle k-30
\@faxfalse t-20
\@faxtrue t-20
\@finclfalse f-175
\@fincltrue f-175
\@firstfoot t-521
\@firsthead t-289
\@firstheadfootfield t-683
\@foldfalse t-121
\@foldmark t-570
\@foldmarks t-570
\@foldtrue t-121
\@fontsizefilebase e-21
\@footnotetext h-156
\@fpbot(Länge) e-185
\@fpsep(Länge) e-185

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\@fptop(Länge) e-185
\@fslfalse i-7
\@fsltrue i-7
\@gen@firsthead t-289
\@globaloptionslist b-102
\@headlines f-34
\@hfoldmark t-570
\@hinclfalse f-172
\@hincltrue f-172
\@hslfalse i-1
\@hsltrue i-1
\@idxitem o-49
\@komalongtablefalse g-48
\@komalongtabletrue g-48
\@komavar@err u-63
\@l@number m-181
\@landscapefalse f-492
\@landscapetrue f-492
\@list@extra j-232
\@listI j-239
\@listi e-233, j-239
\@listii e-233, j-239
\@listiii e-233, j-239
\@listiv e-233
\@listv e-233
\@listvi e-233
\@loadwithoptions q-28
\@locfield t-1008
\@logofalse t-23
\@logotrue t-23
\@lowertitleback k-30
\@mainmatterfalse l-204
\@mainmattertrue l-204
\@makecaption g-175
\@makechapterhead l-467
\@makefnmark h-176
\@makeschapterhead l-641
\@maketitle k-195, t-1266
\@maybeautodot l-236
\@minclfalse f-178
\@mincltrue f-178
\@mpfootins(Länge) c-53
\@newkomavar u-1
\@newplength v-1
\@nextfoot t-1362
\@nexthead t-1322
\@nobreakcr t-1414
\@nobreakvspace t-1414
\@nobreakvspacex t-1414
\@normalheadings l-201
\@ogobble t-1007
\@onefilewithoption q-43
\@openbib@code n-36
\@openrightfalse l-1
\@openrighttrue l-1
\@orgdatefalse d-1
\@orgdatetrue d-1
\@pageat t-247
\@part l-337
\@phonefalse t-19
\@phonetrue t-19
\@pnumwidth m-176
\@processto t-1398
\@pt@@scan e-7
\@pt@scan e-7
\@ptsize e-1
\@publishers k-30
\@refwidefalse t-198
\@refwidetrue t-198
\@replacefilewithext q-59
\@restonecolfalse c-23
\@restonecoltrue c-23
\@rulefrom t-14
\@sAfterClass q-78
\@sAfterPackage q-78
\@saddchap l-665
\@saddpart l-437
\@saddsec l-717
\@savevbox t-659
\@savevbox@warningfalse ... t-659
\@savevbox@warningtrue t-659
\@schapter l-641, A-48
\@seascapefalse f-491
\@seascapetrue f-491
\@seccntformat l-312
\@sect l-254
\@set@reffield t-1154
\@set@reffields t-1165
\@setcapindent g-178
\@setcapmargin g-316
\@setif s-82
\@setkomaname u-35
\@setkomavar u-35
\@setplength v-26
\@showenvelope w-575
\@smallheadings l-201
\@spart l-337, A-48

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\@ssect A-48
\@subjafterfalse t-212
\@subjaftertrue t-212
\@subject k-30
\@subjfalse t-212
\@subjfield t-1266
\@subjtrue t-212
\@subtitle k-30
\@tablecaptionabovefalse . . . g-1
\@tablecaptionabovetrue g-1
\@tempdimaismaxwidthof . . . t-1449
\@textsubscript c-107
\@titlehead k-30
\@titlepagefalse k-1
\@titlepagetrue k-1
\@tocleftfalse m-1
\@toclefttrue m-1
\@tocrmarg m-176
\@topnewpage l-467
\@typearea f-541
\@typearea@end f-668
\@uppertitleback k-30
\@usefolds t-122
\@usekomaname u-76
\@usekomavar u-76
\@vfoldmark t-599
\@wwwfalse t-22
\@wwwtrue t-22
\@xproc t-1398
\@xtrimtempa t-1445
10pt (Option) e-1
11pt (Option) e-1
12pt (Option) e-1

A

a4paper (Option) f-396
a5paper (Option) f-396
\abovecaptionskip(Länge) . . . g-55
abstract (Umgebung) k-259
abstract (Option) k-26
\abstractname l-317
abstractoff (Option) k-26
abstracton (Option) k-26
\activateareas f-669
\add@extra@listi j-232
\addchap l-665
\addchaptertocentry l-638
addmargin (Umgebung) p-102
addmargin* (Umgebung) p-102
\addpart l-437
\addparttocentry l-434
\addrchar y-4
\addrentry y-4
addressee (Element mit Schrift) t-464
addresseeimage (Variable) . . t-1510
addrfield (Option) t-24
\addsec l-717
\addtocentrydefault m-240
\addtokomafont e-604
\addtolengthplength v-36
\addtoreffields t-1166
\adrchar y-1
\adrentry y-1
\after@starttoc m-209
\AfterBibliographyPreamble n-203
\AfterCalculatingTypearea . f-665
\AfterClass q-78, *219*
\AfterClass* *219*
\AfterFile q-78, *219*
\AfterKOMAoptions b-647
\AfterPackage q-78, *219*
\AfterPackage* *219*
\AfterReadingMainAux . q-203, *222*
\aliaskomafont e-673
\appendix l-216
\appendixmore l-216
\appendixname l-320
appendixprefix (Option) l-67
\areaset f-954
\arraycolsep(Länge) p-27
\arrayrulewidth(Länge) p-27
\AtBeginLetter t-1411
\AtEndBibliography n-203
\AtEndLetter t-1413
\autodot l-263

B

b5paper (Option) f-396
\backaddr@format t-78
backaddress (Element mit Schrift) t-458
backaddress (Variable) t-1510
backaddress (Option) t-78
backaddressseparator (Variable) t-1529
\backmatter l-213
BCOR (Option) f-180

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\before@starttoc m-179
\BeforeClass q-78, *219*
\BeforeClosingMainAux . q-203, *221*
\BeforeFile q-78, *219*
\BeforePackage q-78, *219*
\belowcaptionskip(Länge) ... g-55
\bf e-560
\bib@after@hook n-193
\bib@before@hook n-193
\bib@beginhook n-34
\bib@endhook n-34
\bib@heading n-52
\bibindent(Länge) n-117
bibliography (Option) n-62
\bibname n-112
\bibpreamble n-114
bibtotoc (Option) n-110
bibtotocnumbered (Option) .. n-110
bigheadings (Option) l-201
\bottomfraction g-516
bottomnumber (Zähler) g-516
\bprot@dottedtocline l-1026
\BreakBibliography .. n-137, n-195

C

\c@pti@nb@sid@ g-102
\c@pti@nb@sid@b@x g-101
\cal e-571
\cap@left g-233
\cap@margin g-314
\cap@right g-233
\cap@width g-233
\capfont g-171
\caplabelfont g-171
\caption g-55
\captionabove g-61
\captionbelow g-61
captionbeside (Umgebung) ... g-65
\captionformat g-59
\captionof g-526
captions (Option) g-5
\captionsamerican d-144
\captionsaustrian d-144
\captionsbritish d-144
\captionscroation d-144
\captionsdutch d-144
\captionsenglish d-144
\captionsfinnish d-144
\captionsfrench d-144
\captionsgerman d-144
\captionsitalian d-144
\captionsngerman d-144
\captionsnorsk d-144
\captionsspanish d-144
\captionsUKenglish d-144
\captionsUSenglish d-144
\cc t-1482
ccseparator (Variable) t-1529
\changefontsizes e-284
\chapapp l-321
\chapappifchapterprefix ... l-321
\chapappifprefix l-321
\chapter l-467
chapter (Zähler) l-266
\chapter@o@preamble l-899
\chapter@u@preamble l-899
chapteratlists (Option) ... m-148
chapterentry (Element mit Schrift) l-776
chapterentrypagenumber (Element mit Schrift) l-778
\chapterformat l-266
\chapterheadendvskip l-85
\chapterheadstartvskip l-85
\chaptermark l-230
\chaptermarkformat l-266
\chaptername l-317
\chapterpagestyle i-264
chapterprefix (Option) l-64
\ClassInfoNoLine b-475
\ClassName a-24
cleardoubleempty (Option) ... i-15
\cleardoubleemptypage i-15
\cleardoubleevenemptypage .. i-15
\cleardoubleevenpage i-15
\cleardoubleevenplainpage .. i-15
\cleardoubleevenstandardpage i-15
\cleardoubleevenusingstyle . i-15
\cleardoubleoddemptypage ... i-15
\cleardoubleoddpage i-15
\cleardoubleoddplainpage ... i-15
\cleardoubleoddstandardpage i-15
\cleardoubleoddusingstyle .. i-15
\cleardoublepage i-15
cleardoublepage (Option) i-15
cleardoubleplain (Option) ... i-15

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\cleardoubleplainpage i-15
cleardoublestandard (Option) i-15
\cleardoublestandardpage ... i-15
\cleardoubleusingstyle i-15
\closing t-1442
\columnsep(Länge) j-129
\columnseprule(Länge) j-129
\contentsname m-220
customer (Variable) t-1549

D

date (Variable) t-1549
\dateamerican d-144
\dateaustrian d-144
\datebritish d-144
\datecroatian d-144
\datedutch d-144
\dateenglish d-144
\datefinnish d-144
\datefrench d-144
\dategerman d-144
\dateitalian d-144
\datengerman d-144
\datenorsk d-144
\datespanish d-144
\dateUKenglish d-144
\dateUSenglish d-144
\dblfloatpagefraction g-516
\dblfloatsep(Länge) e-185
\dbltopfraction g-516
dbltopnumber (Zähler) g-516
\dedication k-30
\default@par@update j-18
\defaultreffields t-1172
\deffootnote h-100
\deffootnotemark h-131
\DefineFamily b-43
\DefineFamilyKey b-43
\DefineFamilyMember b-43
\descfont p-141
description (Umgebung) p-47
\dictum c-32
dictum (Element mit Schrift) .. c-32
dictumauthor (Element mit Schrift) c-32
\dictumauthorformat c-32
dictumtext (Element mit Schrift) c-32
\dictumwidth c-32
DIV (Option) f-186
DIVcalc (Option) f-200
DIVclassic (Option) f-200
\dont@let@as@internal@defined b-178
\doublerulewidth(Länge) p-27
draft (Option) c-1
dvips (Option) f-390

E

\edgesize w-421
Elemente mit Schrift:
PPdata t-463
PPlogo t-462
addressee t-464
backaddress t-458
chapterentrypagenumber . l-778
chapterentry l-776
dictumauthor c-32
dictumtext c-32
dictum c-32
envelope w-726
field w-384
foldmark t-454
footnoterule h-87
fromaddress t-456
fromname t-455
fromrule t-457
labelinglabel p-143
labelingseparator p-143
letter w-727
measure w-812
minisec l-757, l-784
pagefoot i-84
pageheadfoot i-84
pagehead i-84
pagenumber i-84
pagination i-84
partentrypagenumber l-775
partentry l-774
prioritykey t-461
priority t-460
refname t-467
refvalue t-468
sectionentrypagenumber . l-782
sectionentry l-780
specialmail t-459
subject t-474
subtitle k-256
title t-473

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

toaddress t-466
toname t-465
emailseparator (Variable) . . t-1529
\encl t-1496
enclseparator (Variable) . . . t-1529
enlargefirstpage (Option) . . t-245
envelope (Element mit Schrift) w-726
executivepaper (Option) f-396
\ext@figure g-548
\ext@table g-578
extendedfeature (Option) z-35
\extratitle k-30

F

\f@parfillskip j-10
\f@parindent j-10
\f@parskip j-10
\Family@@String@@Key b-809
\Family@Numerical@Key b-756
\Family@Option b-581
\Family@String@Key b-809
\FamilyBoolKey b-718
\FamilyElseValues b-715
\FamilyExecuteOptions b-480
\FamilyNumericalKey b-756
\FamilyOption b-581
\FamilyOptions b-513
\FamilyProcessOptions b-111
\FamilySetBool b-725
\FamilySetNumerical b-771
\FamilyStringKey b-809
\FamilyUnknownKeyValue b-699
faxseparator (Variable) t-1529
\fboxrule(Länge) c-53
\fboxsep(Länge) c-53
field (Element mit Schrift) . w-384
figure (Zähler) g-530
figure (Umgebung) g-530
\figureformat g-530
\figurename g-528
final (Option) c-1
\firstfoot t-521
firstfoot (Variable) t-521
firstfoot (Option) t-246
\firsthead t-289
firsthead (Variable) t-289
firsthead (Option) t-1
fleqno (Option) c-12
float (Option) A-152
\float@@listhead m-40
\float@addtolists . . A-164, A-265
\float@listhead m-247
\floatc@komascript g-503
\floatpagefraction g-516
\floatsep(Länge) e-185
\flushbottom t-1408
\FN@mf@check h-137
\fnum@figure g-549
\fnum@table g-579
foldmark (Element mit Schrift) t-454
foldmarks (Option) t-121
fontsize (Option) e-22
footexclude (Option) f-175
\footfont i-81
footinclude (Option) f-175
\footincludeoff i-74
\footincludeon i-74
\footins(Länge) e-185
footnosepline (Option) i-7
footnote (Zähler) h-99
\footnoterule h-46
footnoterule (Element mit Schrift) h-87
footnotes (Option) h-1
\footnotesep(Länge) e-185
\footnotesize e-69, j-239
\footref h-180
footsepline (Option) i-7
\footseplineoff i-74
\footseplineon i-74
\fps@figure g-546
\fps@table g-576
fromaddress (Element mit Schrift) t-456
fromaddress (Variable) t-1549
fromalign (Option) t-2
frombank (Variable) t-1549
fromemail (Variable) t-1549
fromemail (Option) t-21
fromfax (Variable) t-1549
fromfax (Option) t-20
fromlogo (Variable) t-1510
fromlogo (Option) t-23
fromname (Element mit Schrift) t-455
fromname (Variable) t-1549
fromphone (Variable) t-1549
fromphone (Option) t-19

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

fromrule (Element mit Schrift) t-457
fromrule (Option) t-14
fromurl (Variable) t-1549
fromurl (Option) t-22
fromzipcode (Variable) t-1510
\frontmatter l-205
\fs@komaabove g-503
\fs@komabelow g-503
\ftn@font h-125
\ftn@rule@height h-89
\ftn@rule@test@values h-55
\ftn@rule@width h-88
\ftnm@font h-125
\ftntm@font h-134
\ftype@figure g-547
\ftype@table g-577

G

\g@addnumerical@date d-128
\gobble@tillscr@@next b-318

H

halfparskip (Option) j-22
halfparskip* (Option) j-22
halfparskip+ (Option) j-22
halfparskip- (Option) j-22
headexclude (Option) f-172
\headfont i-81
headheight (Option) f-166
headinclude (Option) f-172
\headincludeoff i-74
\headincludeon i-74
headings (Option) l-87
headlines (Option) f-160
headnosepline (Option) i-1
headsepline (Option) i-1
\headseplineoff i-74
\headseplineon i-74
\Huge e-69
\huge e-69
hyperref (Option) A-28

I

\idx@@heading o-1, o-5
\idx@heading o-26
idxtotoc (Option) o-22
\if@abstrt k-26
\if@afield t-24
\if@altsecnumformat l-231
\if@at@twocolumn l-467
\if@atdocument b-207
\if@autodot l-231
\if@baddr t-78
\if@bib@break@ n-190
\if@bigloc t-107
\if@capbreak g-177
\if@captionabove g-60
\if@chapterprefix l-64
\if@chaptertolists m-148
\if@dynlist m-44
\if@email t-21
\if@enlargefp t-245
\if@fax t-20
\if@fincl f-175
\if@firstfoot t-246
\if@firsthead t-1
\if@fold t-121
\if@fsl i-7
\if@hincl f-172
\if@hsl i-1
\if@komalongtable g-48
\if@komavarempty u-99
\if@landscape f-492
\if@logo t-23
\if@mainmatter l-204
\if@mincl f-178
\if@openright l-1
\if@orgdate d-1
\if@phone t-19
\if@refwide t-198
\if@restonecol c-23
\if@savevbox@warning t-659
\if@seascape f-491
\if@semitwoside f-202
\if@subj t-212
\if@subjafter t-212
\if@tablecaptionabove g-1
\if@titlepage k-1
\if@tocleft m-1
\if@twoside f-202, i-13
\if@www t-22
\ifdimen b-321
\ifdofullc@p g-175
\ifdvioutput b-435
\ifkomavar u-92
\ifkomavarempty u-99
\ifnotundefined b-274
\ifnumber b-293

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\ifonelinecaptions g-2
\ifpdfoutput b-391
\ifpdftex b-381
\ifpsoutput b-413
\ifscap@alt g-69
\ifscr@key@found@ b-571
\ifstr b-283
\ifthispageodd c-71
\ifthispagewasodd c-98
\ifundefinedorrelax b-271
\ifVTeX b-387
index (Option) o-5
\index@preamble o-23
\indexname o-56
\indexpagestyle i-264
\indexspace o-49
\InputIfFileExists q-6
internalonly (Option) b-178
\intextsep(Länge) e-185
invoice (Variable) t-1549
\is@thispageodd c-73
\iso@@paper f-105
\iso@paper f-105
\isopaper f-46
\it e-560

K

\KOMA@@@DeclareDeprecatedOption b-822
\KOMA@@@DeclareObsoleteOption b-874
\KOMA@@@DeclareStandardOption b-848
\KOMA@@DeclareDeprecatedOption b-822
\KOMA@@DeclareObsoleteOption b-874
\KOMA@@DeclareStandardOption b-848
\KOMA@DeclareDeprecatedOption b-822
\KOMA@DeclareObsoleteOption b-874
\KOMA@DeclareStandardOption b-848
\KOMA@ifkey b-747
\KOMA@iv@DeclareObsoleteOption b-874
\KOMA@key b-96
\KOMA@key@familylist b-96
\KOMA@ncmdkey b-800
\KOMA@newobsoletecommand .. b-911
\KOMA@options b-645
\KOMA@set@ifkey b-753
\KOMA@set@ncmdkey b-806
\KOMA@unknown@keyval b-744
\KOMA@UseDeprecatedOption . b-834
\KOMA@UseObsoleteCommand .. b-915
\KOMA@UseObsoleteOption ... b-897
\KOMA@UseStandardOption ... b-860
\KOMAClassName a-24
\KOMAExecuteOptions b-509
\KOMAoption b-680
\KOMAoptions b-627
\KOMAProcessOptions b-174
\KOMAScript r-1
\KOMAScriptVersion a-1

L

\l@addto@macro b-664
\l@chapter l-945
\l@chapteratlist l-978
\l@figure m-286
\l@paragraph l-1014
\l@part l-909
\l@section l-979
\l@subparagraph l-1014
\l@subsection l-1014
\l@subsubsection l-1014
\l@table m-288
\labelenumi p-32
\labelenumii p-32
\labelenumiii p-32
\labelenumiv p-32
labeling (Umgebung) p-56
\labelinglabel p-56
labelinglabel (Element mit Schrift) p-143
labelingseparator (Element mit Schrift) p-143
\labelitemi p-32
\labelitemii p-32
\labelitemiii p-32
\labelitemiv p-32
\labelsep(Länge) p-1
\labelwidth(Länge) p-1
landscape (Option) f-396
\LARGE e-69
\Large e-69

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\large e-69
\last@l@number m-179
\lco@test w-1
\LCOWarning w-116, x-10
\LCOWarningNoLine ... w-116, x-10
\leftmargin(Länge) p-1
\leftmargini(Länge) p-1
\leftmarginii(Länge) p-1
\leftmarginiii(Länge) p-1
\leftmarginiv(Länge) p-1
\leftmarginv(Länge) p-1
\leftmarginvi(Länge) p-1
legalpaper (Option) f-396
leqno (Option) c-12
\let@as@internal@defined .. b-213
letter (Umgebung) t-1387
letter (Element mit Schrift) w-727
\LetterOptionNeedsPapersize w-51
letterpaper (Option) f-396
\lineskip(Länge) j-127
\listfigurename m-279
listings (Option) A-253
\listof A-164
listof (Option) m-45
\listoffigures m-285
\listoflofname m-279
\listoftables m-287
\listoftocname m-220
\listotlofname m-279
listsindent (Option) m-146
listsleft (Option) m-146
liststotoc (Option) m-144
liststotocnumbered (Option) m-145
\listtablename m-279
\LoadLetterOption w-14
location (Variable) t-1510
locfield (Option) t-107
\LOPNP@option w-51
\LOPNP@size w-51
\lowertitleback k-30
\lstlistoflistings A-265
\LT@array g-443
\LT@caption g-443
\LT@makecaption g-443
\ltr@len@@firstfootvpos ... t-509
\ltr@len@backaddrheight ... t-489
\ltr@len@bfoldmarklength .. t-481
\ltr@len@bfoldmarkvpos t-477
\ltr@len@firstfoothpos t-511
\ltr@len@firstfootwidth ... t-510
\ltr@len@firstheadhpos t-515
\ltr@len@firstheadvspos ... t-513
\ltr@len@firstheadwidth ... t-514
\ltr@len@foldmarkhpos t-475
\ltr@len@foldmarkthickness t-486
\ltr@len@foldmarkvpos t-476
\ltr@len@fromrulethickness t-519
\ltr@len@fromrulewidth t-517
\ltr@len@hfoldmarklength .. t-481
\ltr@len@lfoldmarkhpos t-480
\ltr@len@lfoldmarklength .. t-481
\ltr@len@locheight t-502
\ltr@len@lochpos t-501
\ltr@len@locvpos t-499
\ltr@len@locwidth t-500
\ltr@len@mfoldmarklength .. t-481
\ltr@len@mfoldmarkvpos t-477
\ltr@len@pfoldmarklength .. t-481
\ltr@len@PPdatamatrixvskip t-496
\ltr@len@PPheadheight t-494
\ltr@len@PPheadwidth t-495
\ltr@len@refaftervskip t-506
\ltr@len@refhpos t-505
\ltr@len@refvpos t-503
\ltr@len@refwidth t-504
\ltr@len@sigbeforevskip ... t-507
\ltr@len@sigindent t-508
\ltr@len@specialmailindent t-491
\ltr@len@specialmailrightindent t-492
\ltr@len@subjectaftervskip t-1318
\ltr@len@subjectbeforevskip t-1320
\ltr@len@subjectvpos t-1316
\ltr@len@tfoldmarkvpos t-477
\ltr@len@toaddrheight t-498
\ltr@len@toaddrhpos t-490
\ltr@len@toaddrindent t-493
\ltr@len@toaddrvpos t-488
\ltr@len@toaddrwidth t-497
\ltr@priority t-68

M

\mainmatter l-209
\maketitle k-72
\marginline g-52
\maybe@pair f-493

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

measure (Element mit Schrift) w-812
\measuredFrameLB w-734
\measuredFrameLT w-734
\measuredFrameRB w-734
\measuredFrameRT w-734
\measureLineH w-789
\measureLineHB w-796
\measureLineHT w-796
\measureLineV w-758
\measureLineVL w-766
\measureLineVR w-766
\minisec l-757
minisec (Element mit Schrift) l-757, l-784
\mit e-571
\move@frompl t-557
\move@frompt t-545
\move@topl t-552
\move@topt t-539
mpexclude (Option) f-178
mpinclude (Option) f-178
\multfootsep h-171
\multiplefootnotemarker h-173
\multiplefootnoteseparator h-168
myref (Variable) t-1549

N

\new@tpo@label c-71
\newbibstyle n-1
\newblock n-36
\newcaptionname d-9
\newfloat A-164
\newkomafont e-662
\newkomavar u-1
\next@tpage k-71
\nextfoot t-1362
nextfoot (Variable) t-1362
\nexthead t-1322
nexthead (Variable) t-1322
\no@bib@break n-190
noappendixprefix (Option) l-67
nochapterprefix (Option) l-64
noonelinecaption (Option) g-50
normalheadings (Option) l-201
\normallineskip(Länge) j-127
\normalsize e-69
notitlepage (Option) k-1
\numberline m-167
\numberline@box m-175
\numberline@numberformat m-174
\numberline@postfix m-173
\numberline@prefix m-172
numbers (Option) l-23
numericaldate (Option) d-1

O

onecolumn (Option) f-233
onelinecaption (Option) g-50
\onelinecaptionsfalse g-2
\onelinecaptionstrue g-2
oneside (Option) f-202
open (Option) l-1
openany (Option) l-21
openbib (Option) n-109
\opening t-1425
openright (Option) l-21
Optionen:
10pt e-1
11pt e-1
12pt e-1
BCOR f-180
DIVcalc f-200
DIVclassic f-200
DIV f-186
a4paper f-396
a5paper f-396
abstractoff k-26
abstracton k-26
abstract k-26
addrfield t-24
appendixprefix l-67
b5paper f-396
backaddress t-78
bibliography n-62
bibtotocnumbered n-110
bibtotoc n-110
bigheadings l-201
captions g-5
chapteratlists m-148
chapterprefix l-64
cleardoubleempty i-15
cleardoublepage i-15
cleardoubleplain i-15
cleardoublestandard i-15
draft c-1
dvips f-390
enlargefirstpage t-245
executivepaper f-396

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

extendedfeature z-35
final c-1
firstfoot t-246
firsthead t-1
fleqno c-12
float A-152
foldmarks t-121
fontsize e-22
footexclude f-175
footinclude f-175
footnosepline i-7
footnotes h-1
footsepline i-7
fromalign t-2
fromemail t-21
fromfax t-20
fromlogo t-23
fromphone t-19
fromrule t-14
fromurl t-22
halfparskip* j-22
halfparskip+ j-22
halfparskip- j-22
halfparskip j-22
headexclude f-172
headheight f-166
headinclude f-172
headings l-87
headlines f-160
headnosepline i-1
headsepline i-1
hyperref A-28
idxtotoc o-22
index o-5
internalonly b-178
landscape f-396
legalpaper f-396
leqno c-12
letterpaper f-396
listings A-253
listof m-45
listsindent m-146
listsleft m-146
liststotocnumbered m-145
liststotoc m-144
locfield t-107
mpexclude f-178
mpinclude f-178
noappendixprefix l-67
nochapterprefix l-64
noonelinecaption g-50
normalheadings l-201
notitlepage k-1
numbers l-23
numericaldate d-1
onecolumn f-233
onelinecaption g-50
oneside f-202
openany l-21
openbib n-109
openright l-21
open l-1
orgdate d-7
origlongtable g-48
pagenumber t-247
pagesize f-246
paper f-404
parindent j-22
parskip* j-22
parskip+ j-22
parskip- j-22
parskip j-22
pdftex f-393
pointednumbers l-62
pointlessnumbers l-62
portrait f-396
priority t-68
refline t-198
scrdate d-7
slocfield t-119
smallheadings l-201
subject t-210
tablecaptionabove g-46
tablecaptionbelow g-46
titlepage k-1
tocindent m-38
tocleft m-38
toc m-2
twocolumn f-233
twoside f-202
version s-1
wlocfield t-119
orgdate (Option) d-7
origlongtable (Option) g-48
\othersectionlevelsformat . l-312

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

P

\p@enumii p-32
\p@enumiii p-32
\p@enumiv p-32
\PackageInfoNoLine b-470
pagefoot (Element mit Schrift) . i-84
pagehead (Element mit Schrift) . i-84
pageheadfoot (Element mit Schrift) i-84
\pagemark i-103
pagenumber (Element mit Schrift) i-84
pagenumber (Option) t-247
pagesize (Option) f-246
pagination (Element mit Schrift) i-84
paper (Option) f-404
\par@update j-18
\par@updaterelative j-146
\paragraph l-724
paragraph (Zähler) l-266
\parfillskip@update j-171
\parheademptypage l-337
parindent (Option) j-22
\parindent@update j-171
parskip (Option) j-22
parskip* (Option) j-22
parskip+ (Option) j-22
parskip- (Option) j-22
\parskip@update j-171
\part l-337
part (Zähler) l-266
\part@preamble@o l-904
\part@preamble@u l-904
partentry (Element mit Schrift) l-774
partentrypagenumber (Element mit Schrift) l-775
\partformat l-266
\partheadendvskip l-337
\partheadmidvskip l-337
\partheadstartvskip l-337
\partmark l-465
\partname l-317
\partopsep(Länge) e-185
\partpagestyle i-264
pdftex (Option) f-393
phoneseparator (Variable) . . t-1529
place (Variable) t-1510
placeseparator (Variable) . . t-1529
\pnumfont i-81
pointednumbers (Option) l-62
pointlessnumbers (Option) . . . l-62
portrait (Option) f-396
\post@kopt@hook b-647
PPcode (Variable) t-1510
PPdata (Element mit Schrift) . t-463
PPdatamatrix (Variable) t-1510
PPlogo (Element mit Schrift) . t-462
\presig@skip t-1442
\PreventPackageFromLoading q-112, *221*
priority (Element mit Schrift) t-460
priority (Option) t-68
prioritykey (Element mit Schrift) t-461
\protected@immediate@write q-278
\providecaptionname d-9
\ps t-1477
\ps@headings i-74
\ps@myheadings i-74
\ps@plain i-74
\publishers k-30

Q

quotation (Umgebung) p-86
quote (Umgebung) p-86

R

\raggedchapterentry l-977
\raggeddictum c-32
\raggeddictumauthor c-32
\raggeddictumtext c-32
\raggedpart l-768
\raggedsection l-766
\raggedsectionentry l-1009
\raggedsubject t-211
\recalctypearea f-953
refline (Option) t-198
\refname n-112
refname (Element mit Schrift) . t-467
refvalue (Element mit Schrift) t-468
\removereffields t-1169
\renewcaptionname d-9
\ReplaceClass q-59, *220*
\ReplaceInput q-56, *219*
\ReplacePackage q-59, *220*
\ResetPreventPackageFromLoading q-163, *221*
\rm e-560

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

S

\sbltextfloatsep(Länge) ... e-185
\sc ... e-560
\scap@altfalse ... g-69
\scap@alttrue ... g-69
\scap@caption ... g-65
\scap@currentlabel ... g-68
\scap@move ... g-67
\scap@pos ... g-3
\scap@valign ... g-4
\scap@width ... g-66
\scr@@@defaultunits ... j-201
\scr@@defaultunits ... j-201
\scr@@loadwithoptions ... q-28
\scr@@onefilewithoption ... q-43
\scr@altsecnumhook ... l-54
\scr@altsecnumhookfalse ... l-54
\scr@altsecnumhooktrue ... l-54
\scr@bibstyle@ ... n-33
\scr@bibstyle@oldstyle ... n-39
\scr@bibstyle@openstyle ... n-39
\scr@bibstylelist ... n-36
\scr@caption ... g-61
\scr@caption@before@longtable@warning ... g-443
\scr@compatibility ... s-1
\scr@defaultunits ... j-201
\scr@do@hack@listings ... A-265
\scr@dotchangeatdocument ... l-23
\scr@dottedtocline ... m-179
\scr@ext@activate ... z-14
\scr@ext@activateable ... z-14
\scr@ext@addto@activateable z-14
\scr@firsthead@postsetvar . t-312
\scr@float@addtolists@warning ... m-268
\scr@float@listhead@warning m-259
\scr@fnt@caption ... g-173
\scr@fnt@captionlabel ... g-173
\scr@fnt@chapter ... l-784
\scr@fnt@chapterprefix ... l-784
\scr@fnt@descriptionlabel . p-142
\scr@fnt@disposition ... l-784
\scr@fnt@footnote ... h-125
\scr@fnt@footnotenlabel ... h-125
\scr@fnt@footnotenumber ... h-125
\scr@fnt@footnoteref ... h-134
\scr@fnt@footnotereference h-134
\scr@fnt@instead@footnotetext ... h-125
\scr@fnt@paragraph ... l-784
\scr@fnt@part ... l-784
\scr@fnt@partnumber ... l-784
\scr@fnt@section ... l-784
\scr@fnt@sectioning ... l-784
\scr@fnt@subject ... k-257
\scr@fnt@subparagraph ... l-784
\scr@fnt@subsection ... l-784
\scr@fnt@subsubsection ... l-784
\scr@fnt@title ... k-257
\scr@fnt@wrn@pagefoot ... i-84
\scr@fnt@wrn@pagehead ... i-84
\scr@footmisc@options ... h-29
\scr@footnotemark ... h-31
\scr@gobbleopt ... q-202
\scr@hack@load ... A-12
\scr@hook@acma ... q-257
\scr@hook@bcma ... q-257
\scr@if@family ... b-43
\scr@if@familymember ... b-43
\scr@ifdvioutput ... b-435
\scr@ifexpected ... A-1
\scr@ifpdfoutput ... b-391
\scr@ifpdftex ... b-381
\scr@ifpsoutput ... b-413
\scr@ifundefinedorrelax ... b-30
\scr@ifVTeX ... b-387
\scr@key@atlist ... b-571
\scr@key@name ... b-571
\scr@key@split@name@value . b-571
\scr@key@value ... b-571
\scr@load@hook ... q-71
\scr@loadwithoptions ... q-28
\scr@mf@prepare ... h-145
\scr@new@selectfont ... j-133
\scr@numberline ... m-179
\scr@onefilewithoption ... q-43
\scr@optiontest ... b-167
\scr@parfillskip ... j-22
\scr@parindent ... j-22
\scr@parskip ... j-22
\scr@pti@nerr@r ... f-155
\scr@pti@ntest ... b-167
\scr@redefine@newpage ... q-225

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\scr@removechar t-190
\scr@replacefile@msg q-68
\scr@restore@newpage q-225
\scr@saved@checkend q-223
\scr@saved@dofilelist q-223
\scr@saved@footnotemark h-31
\scr@saved@footnotetext ... h-156
\scr@saved@newpage q-225
\scr@sect l-254
\scr@selectfont j-133
\scr@set@load@hook q-75
\scr@smashdp b-457
\scr@sr@co b-909
\scr@ta@compatibility s-1
\scr@topnewpage l-467
\scr@tpo c-72
\scr@updatedvipspagesize .. f-273
\scr@v@2.9 ..... s-43
\scr@v@2.95 ..... s-43
\scr@v@2.95a s-43
\scr@v@2.95b s-43
\scr@v@2.96 ..... s-43
\scr@v@2.96a s-43
\scr@v@2.97 ..... s-43
\scr@v@2.97a s-43
\scr@v@2.97b s-43
\scr@v@2.97c s-43
\scr@v@2.97d s-43
\scr@v@2.97e s-43
\scr@v@2.98 ..... s-43
\scr@v@2.98a s-43
\scr@v@2.98b s-43
\scr@v@2.98c s-43
\scr@v@2.9t s-43
\scr@v@3.00 ..... s-43
\scr@v@3.01 ..... s-43
\scr@v@3.01a s-43
\scr@v@3.01b s-43
\scr@v@3.01c s-43
\scr@v@3.02 ..... s-43
\scr@v@3.02b s-43
\scr@v@3.02c s-43
\scr@v@3.03 ..... s-43
\scr@v@3.03a s-43
\scr@v@3.03b s-43
\scr@v@3.04 ..... s-43
\scr@v@3.04a s-43
\scr@v@3.05 ..... s-43
\scr@v@3.05a s-43
\scr@v@3.06 ..... s-43
\scr@v@3.07 ..... s-43
\scr@v@3.08 ..... s-43
\scr@v@3.08a s-43
\scr@v@first s-43
\scr@v@last s-43
scrdate (Option) d-7
\scriptsize e-69
\scrlfile@excludedpackages q-112
\scrlfile@RequirePackage .. q-177
\scrlfile@saved@RequirePackage q-112
secnumdepth (Zähler) l-264
\sectfont l-771
\section l-724
section (Zähler) l-266
sectionentry (Element mit Schrift) l-780
sectionentrypagenumber (Element mit Schrift) l-782
\sectionmarkformat l-266
\selectfont j-133
\set@@@@preamble l-803
\set@@@preamble l-803
\set@@preamble l-803
\set@dvipdfmxpagesize f-273
\set@dvipspagesize f-273
\set@l@number m-181
\set@pdftexpagesize f-273
\set@preamble l-803
\set@preamble@hpos l-863
\set@preamble@vpos l-881
\set@tempdima@hw i-74
\set@vtexpagesize f-273
\setbibpreamble n-114
\setcaphanging g-208
\setcapindent g-178
\setcapmargin g-316
\setcapwidth g-236
\setchapterpreamble l-899
\SetDIVList f-9
\setfootnoterule h-90
\setindexpreamble o-23
\setkomafont e-575
\setkomavar u-35
\setlengthtoplength v-23
\setnot@dvipspagesize f-273

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\setparsizes j-1
\setpartpreamble l-904
\setpresigskip t-1442
\setshowstyle w-431
\SetXDIVList f-10
\sf e-560
\sfb e-560
\showenvelope w-575
\showfield w-385
\showfield@address w-491
\showfield@by@edges w-406
\showfield@by@frame w-385
\showfield@by@rule w-422
\showfield@foot w-468
\showfield@head w-445
\showfield@location w-501
\showfield@refline w-537
\showfield@test w-442
\showfields w-362
\showISOenvelope w-813
\showUScheck w-851
\showUScommercial w-840
\showUSletterCixDW w-854
signature (Variable) t-1510
\size@chapter l-76
\size@chapterprefix l-76
\size@paragraph l-76
\size@part l-76
\size@partnumber l-76
\size@section l-76
\size@subparagraph l-76
\size@subsection l-76
\size@subsubsection l-76
\sl e-560
slocfield (Option) t-119
\small e-69, j-239
smallheadings (Option) l-201
specialmail (Element mit Schrift) t-459
specialmail (Variable) t-1510
\specialmail@format t-1005
\startbreaks t-1414
\stopbreaks t-1414
\stopletter t-1398
\StorePreventPackageFromLoading q-163, *221*
\subitem o-49
\subject k-30
subject (Element mit Schrift) . t-474
subject (Variable) t-1549
subject (Option) t-210
\subject@font k-254
\subject@format t-210
subjectseparator (Variable) t-1529
\subparagraph l-724
subparagraph (Zähler) l-266
\subsection l-724
subsection (Zähler) l-266
\subsectionmarkformat l-266
\subsubitem o-49
\subsubsection l-724
subsubsection (Zähler) l-266
\subtitle k-30
subtitle (Element mit Schrift) k-256

T

\ta@bcor(Länge) f-11
\ta@div f-12
\ta@divfor f-692
\ta@divlist f-522
\ta@divlistelse f-522
\ta@hblk(Länge) f-538
\ta@layoutinfo f-918
\ta@modiv f-628
\ta@temp(Länge) f-538
\ta@temp@goodwidth f-647
\ta@vblk(Länge) f-538
\tabbingsep(Länge) p-27
\tabcolsep(Länge) p-27
table (Zähler) g-560
table (Umgebung) g-560
tablecaptionabove (Option) . . g-46
tablecaptionbelow (Option) . . g-46
\tableformat g-560
\tablename g-528
\tableofcontents m-237
\test@@ifdimunt b-360
\test@dimen b-331
\test@number b-303
\test@posdimen b-336
\test@trueunt b-351
\testdimunt b-351
\textfloatsep(Länge) e-185
\textfraction g-516
\textsubscript c-107
thebibliography (Umgebung) n-119
\thechapter l-266

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\theenumi p-32
\theenumii p-32
\theenumiii p-32
\theenumiv p-32
\theequation c-58
\thefigure g-530
\thefootnotemark h-174
theindex (Umgebung) o-31
\theparagraph l-266
\thepart l-266
\thesection l-266
\thesubparagraph l-266
\thesubsection l-266
\thesubsubsection l-266
\thetable g-560
\thispagewasoddfalse c-98
\thispagewasoddtrue c-98
\tiny e-69
title (Element mit Schrift) . . t-473
title (Variable) t-1510
\titlefont k-254
\titlehead k-30
titlepage (Umgebung) k-237
titlepage (Option) k-1
\titlepagestyle i-264
toaddress (Element mit Schrift) t-466
toaddress (Variable) t-1549
toc (Option) m-2
\toc@heading m-224
\toc@l@number m-236
tocdepth (Zähler) m-222
tocindent (Option) m-38
tocleft (Option) m-38
toname (Element mit Schrift) . t-465
toname (Variable) t-1549
\topfraction g-516
topnumber (Zähler) g-516
totalnumber (Zähler) g-516
\tt e-560
twocolumn (Option) f-233
twoside (Option) f-202
\typearea f-541

U

Umgebungen:
abstract k-259
addmargin* p-102
addmargin p-102
captionbeside g-65
description p-47
figure g-530
labeling p-56
letter t-1387
quotation p-86
quote p-86
table g-560
thebibliography n-119
theindex o-31
titlepage k-237
verse p-74
\unitfactor w-780
\unmeasuredFrame w-727
\uppertitleback k-30
urlseparator (Variable) t-1529
\use@chapter@o@preamble . . . l-473
\use@preamble l-797
\usekomafont e-635
\usekomavar u-76
\useplength v-16

V

\ValPerUnit w-780
Variablen:
PPcode t-1510
PPdatamatrix t-1510
addresseeimage t-1510
backaddressseparator . . t-1529
backaddress t-1510
ccseparator t-1529
customer t-1549
date t-1549
emailseparator t-1529
enclseparator t-1529
faxseparator t-1529
firstfoot t-521
firsthead t-289
fromaddress t-1549
frombank t-1549
fromemail t-1549
fromfax t-1549
fromlogo t-1510
fromname t-1549
fromphone t-1549
fromurl t-1549
fromzipcode t-1510
invoice t-1549
location t-1510

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

myref t-1549
nextfoot t-1362
nexthead t-1322
phoneseparator t-1529
placeseparator t-1529
place t-1510
signature t-1510
specialmail t-1510
subjectseparator t-1529
subject t-1549
title t-1510
toaddress t-1549
toname t-1549
urlseparator t-1529
yourmail t-1549
yourref t-1549
zipcodeseparator t-1529
\vb@t@z t-562
verse (Umgebung) p-74
version (Option) s-1

W

wlocfield (Option) t-119

X

\x@@paper f-46
\x@paper f-46
\XdivY b-197
\XmodY b-202

Y

yourmail (Variable) t-1549
yourref (Variable) t-1549

Z

Zähler:
bottomnumber g-516
chapter l-266
dbltopnumber g-516
figure g-530
footnote h-99
paragraph l-266
part l-266
secnumdepth l-264
section l-266
subparagraph l-266
subsection l-266
subsubsection l-266
table g-560
tocdepth m-222
topnumber g-516
totalnumber g-516
zipcodeseparator (Variable) t-1529

Change History

v1.0
General: Beginn der ersten Version 12
v2.0e
General: bei scrbook gibt es kein einseitiges Layout 137
erste Version von scrartcl, scrreprt und scrbook 10
Konfigurationsdatei typearea.cfg zur Definition weiterer Optionen bzw. Papiergrößen . . . 91
\addchap: \@makechapterhead durch \@makeschapterhead ersetzt 179
\addcontensline durch \addcontentsline ersetzt 179
\afterheadings durch \@afterheading ersetzt . 179
\sectdef durch \secdef ersetzt 179
mehrfach \markboth durch \sectionmark ersetzt . . . 179
\addsec: mehrfach \markboth durch \sectionmark ersetzt 181
footsepline: bei \PassOptionsToPackage fehlte die Paket-Angabe . 135
\ps@headings: im einseitigen Seitenstil \markboth durch \markright ersetzt 139
\ta@divlist: neu (intern) . . . 92
\typearea: Seitenaufteilung wird in die log-Datei geschrieben 104
v2.1a
\addchap: Argument von \chaptermark entfernt bei

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

scrbook 179
\maketitle: das Hilfskonstrukt \@maketitle wurde aus der Unterscheidung herausgenommen (für den Fall, dass \maketitle von einem Paket überladen wird) 154
\SetDIVList: neu 76
v2.1b
\areaset: \headheight von \@ptsize abhängig gemacht 106
\ps@headings: im einseitigen Seitenstil \markboth durch \markright ersetzt 139
v2.1c
\part: Zeichenkette „markboth“ durch Befehl \markboth ersetzt 171
v2.2a
BCOR: neue Option 82
thebibliography: in scrartcl \bibname vollständig durch \refname ersetzt 207
DIV: neue Option 82
\ps@headings: bei scrbook und scrreprt Punkt hinter der section-Nummer entfernt . 138
im einseitigen Seitenstil bei scrartcl und oneside \subsectionmark eingefügt 139
im einseitigen Seitenstil bei scrreprt und oneside \sectionmark eingefügt . 139
\ta@bcor: Definition der Dimension wegen der Option BCOR vorgezogen 76
\ta@div: Definition wegen der Option DIV vorgezogen . . . 76
v2.2b
\addchap: \chaptermark durch \@mkboth ersetzt 179
\addsec: \sectionmark durch \@mkboth ersetzt 181
\closing: \fromsig wird nicht mehr auf \empty getestet sondern direkt eingesetzt . 278
v2.2c
General: Anpassung an CJK von Werner Lemberg. 39
table: Tabellenausgabe auf CJK umgestellt 127
\caption: Fließtitel auf CJK umgestellt 111
\datenorsk: “ hinter \language 55
figure: Abbildungsausgabe auf CJK umgestellt 126
\part: Part-Ausgabe auf CJK umgestellt 171
\ps@headings: Nummern in der Kopfzeile auf CJK umgestellt 138
Punkt hinter der Kapitelnummer in der Kopfzeile entfernt 138
\sfb: nicht mehr mathematisch 72
v2.2d
\appendix: scrartcl benötigt keinen Zähler für \chapter . 166
v2.3
\cal: hier statt im LaTeX-Kern 72
\mit: hier statt im LaTeX-Kern 72
v2.3a
\@mainmatterfalse: \if@mainmatter Anforderung verschoben . 166
\@makecaption: \setbox\tempboxa\hbox durch \sbox\@tempboxa ersetzt 116
Minipage-Flag zurückgesetzt (ist das ok?) 116
General: da bei book seit Version 1.2v die Option oneside wieder ein einseitiges Layout erzeugt, dieses wieder eingebaut 137
abstract: \@endparpenalty zur Verhinderung eines Seitenumbruchs nach dem Abstract-Kopf eingefügt . 159
\chaptermarkformat: Leerraum nach der Kapitelnummer erhöht 169
\footnoterule: \@width verwendet 130
\maketitle: \vfil durch \vfill ersetzt 154
fill durch \fill ersetzt . 154
plus durch \@plus ersetzt 154
Definition von \@makefnmark

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

und \@makefntext unabhängig von math 155
Verwendung von \@makefnmark in \@makefntext 155
\sectionmarkformat: Leerraum nach der Kapitelnummer erhöht 169
\sfb: keine Unterscheidung mehr für den Kompatibilitätsmodus 72
\subsectionmarkformat: Leerraum nach der Kapitelnummer erhöht 169
v2.3b
\@maketitle: Verwendung von \sectfont für den \title 272
\closing: Abstand zwischen Schlussgruß und Signatur erhöht 278
\maketitle: Umbruch bei \@extratitle in der beidseitigen Titelseite korrigiert 154
\next@tpage: \null entfernt. 154
v2.3c
\@seccntformat: CJK erweitert 170
\appendix: alternative Nummerierung bei Verwendung eines Anhangs 166
\captionformat: neu 112
\chapterformat: Duden Regel 6 169
\chaptermarkformat: Duden Regel 6 169
\figureformat: neu 126
\if@altsecnumformat: neuer Schalter 167
\numberline: \numberline umdefiniert 197
\part: alternative Numerierung bei römischer Part-Numerierung 171
\partformat: Duden Regel 6 169
\sectionmarkformat: Duden Regel 6 169
\subsectionmarkformat: Duden Regel 6 169
\tableformat: neu 127
v2.3d
\maketitle: \fill durch fill ersetzt 154
\part: alternative Numerierung korrigiert 171
\subsubsection: wird bei scrartcl nun ebenfalls numeriert und ins Inhaltsverzeichnis geschrieben 181
v2.3e
\@datefield: Abstand zwischen Ort und Datum eingefügt 270
\dateitalian: Ich hoffe, dass das stimmt 48
\maketitle: optionales Argument bei der einseitigen Version erlauben und ignorieren 155
v2.3f
\@makecaption: Fontumschaltung einheitlich für lange und kurze Unter-/Überschriften 116
v2.3g
\@endpart: \@endpart wird für scrartcl nicht mehr definiert 171
\@openbib@code: Neu (intern) 204
abstract: \@beginparpenalty zur Verhinderung eines Seitenumbruchs vor dem Abstract-Kopf eingefügt . 159
\cal: überflüssige Klammerpaare entfernt 72
verse: \item muss keine „[]“-Klammern mehr verarbeiten 215
\l@part: fehlendes Kommentarzeichen ergänzt 188
\maketitle: \@date und \title u. ä. ebenfalls löschen, um Speicher zu sparen 156
\footnote funktioniert nun auch im Titel 154
mehrfach \null bei \next@tpage ergänzt . . . 155
Verwendung von \global\let statt \gdef, um Speicher zu sparen 156
\mit: überflüssige Klammerpaare entfernt 72
\newblock: Grundeinstellung geändert 204

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\next@tpage: \newpage durch \clearpage ersetzt 154
\part: \hbox durch \null ersetzt 171
openbib: \if@openbib (\@openbibtrue, \@openbibfalse) entfernt 206
Definition der openbib Option neu implementiert 206
\subitem: Verwendung von \@idxitem spart Speicher 212
\subsubitem: Verwendung von \@idxitem spart Speicher 212
quotation: \item muss keine „[]“-Klammern mehr verarbeiten 215
quote: \item muss keine „[]“-Klammern mehr verarbeiten 215
v2.3h
General: \relax nach \ProcessOptions ergänzt. 92
Größenbefehle in den Überschriften variabilisiert 171
\addsec: jetzt auch bei scrbook und scrreprt 181
\bib@heading: neu (intern) . 205
idxtotoc: neue Option 211
headings: Größe von \chapter um eine Stufe verringert . 163
Verwendung von \chapterheadstartvskip und \chapterheadendvskip an Stelle von festen vertikalen Abständen am Anfang und am Ende eines Kapitels 163
thebibliography: Verwendung von \bib@heading 207
theindex: Verwendung von \idx@heading 211
bibtotoc: neue Option 206
bibtotocnumbered: neue Option 206
\idx@heading: neu (intern) . 211
bigheadings: neue Option . . 166
listtotoc: neue Option . . . 196
\listoffigures: Verwendung von \lof@heading 201
\listoftables: Verwendung von \lot@heading 201
smallheadings: neue Option 166
pointednumbers: neue Option 162
pointlessnumbers: neue Option 162
normalheadings: neue Option 166
\raggedsection: Überschriften werden \raggedsection gesetzt 183
\raggedsection ist als \raggedright voreingestellt 183
\tableofcontents: Verwendung von \toc@heading . 200
\toc@heading: neu (intern) . 199
v2.4
\maketitle: \footnote funktioniert nun wirklich im Titel 154
v2.4a
\encl: der Doppelpunkt bei den Anlagen wird nur noch gesetzt, wenn \enclname nicht leer ist 279
listtotoc: \listtabelname durch \listtablename ersetzt 196
v2.4b
\@part: \size@partnumer durch \size@partnumber ersetzt 171
v2.4c
\dateamerican: american identisch mit USenglish definiert 48
\datebritish: british identisch mit english definiert 48
\datenorsk: Sprachauswahl um american, british, UKenglish und spanish erweitert 55
\datespanish: Datumsumschaltung nicht mehr zwingend . 48
Ich hoffe, dass das stimmt . . 48
\dateUKenglish: UKenglish identisch mit english definiert 48
\idx@@heading: Kolumnentitel werden nun korrekt behandelt 210
\idx@heading: Kolumnentitel werden nun korrekt behandelt 211

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

v2.4e
\@endpart: Option openany beachten 171
\frontmatter: Option openany beachten 166
\mainmatter: Option openany beachten 166
v2.4f
\ps@headings: \strut in der Kopfzeile eingefügt 138
v2.4g
\addsec: \section* vorgezogen, damit dadurch provozierte Seitenumbrüche im Inhaltsverzeichnis berücksichtigt werden 181
\idx@@heading: Index auch bei scrartcl wieder zweispaltig 210
\idx@heading: Index auch bei scrartcl wieder zweispaltig 211
bigheadings: die Option heißt nun wirklich so; bei Verwendung des alten, falschen Namens wird ein Fehler ausgegeben 166
DIVcalc: neue Option 83
smallheadings: die Option heißt nun wirklich so; bei Verwendung des alten, falschen Namens wird ein Fehler ausgegeben 166
pointednumbers: \@altsecnumformatfalse wird direkt auf \@altsecnumformattrue gesetzt 162
die Option heißt nun wirklich so, bei pointednumber wird explizit ein Fehler gemeldet 162
pointlessnumbers: \@altsecnumformattrue wird direkt auf \@altsecnumformatfalse gesetzt 162
die Option heißt nun wirklich so, bei pointlessnumber wird explizit ein Fehler gemeldet 162
normalheadings: die Option heißt nun wirklich so; bei Verwendung des alten, falschen Namens wird ein Fehler ausgegeben 166
\ta@divfor: Satzspiegel wird nicht mehr direkt berechnet, sondern nur der *DIV*-Wert aus der Defaultliste ermittelt 98
\typearea: überflüssigen Kommentar gelöscht 99
0 versucht den Wert aus der Default-Liste zu verwenden 101
v2.4h
\maketitle: egal ob der Titel auf eine Extraseite kommt oder nicht, \@title wird mit \sectfont gesetzt, wobei die Größe neuerdings danach eingestellt wird 154
v2.4i
\typearea: am Ende von \typearea wird noch \@typeare@end ausgeführt 104
Seitenaufteilung wird etwas später in die log-Datei geschrieben 104
v2.4j
\typearea: bei der Seitenaufteilung wird \textheight nicht mehr doppelt, dafür zusätzlich \baselineskip ausgegeben 104
v2.4k
\listoffigures: \lof@heading nicht nur bei scrartcl 201
\listoftables: \lot@heading nicht nur bei scrartcl 201
v2.4l
\@@makefnmark: neu 132
General: Verwendung der neuen Makros zur Fußnotengestaltgebung 134
\deffootnote: neu 132
\deffootnotemark: neu 133
\maketitle: es ist nicht mehr notwendig \makefntext umzudefinieren, stattdessen wird nur \makefnmark umdefiniert 155
symbolische Fußnoten und Fußnotensymbolbreiten von 0 pt zur besseren Zentrierung

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

des Autors sind auch bei Titelseiten angebracht 154
\marginline: Definition mit Hilfe des optionalen Parameters von \marginpar deutlich vereinfacht 111
\thefootnotemark: neu 134
v2.4m
\labelenumii: öffnende Klammer entsprechend Doku entfernt 214
v2.4n
\@part: für scrartcl \markboth zum Löschen der Kolumnentitel durch \sectionmark ersetzt 171
\@spart: in der Sternvariante \chaptermark bzw. \sectionmark eingefügt, um die Kolumnentitel zu löschen 171
\@typearea: eigentliche Satzspiegelberechnung in \@typearea ausgegliedert . 93
DIVclassic: neue Option 83
\typearea: berechnete Kopfhöhe vom Grundzeilenabstand abhängig 100
optimale *DIV*-Berechnung verbessert 102
optionale, klassische Kreisberechnung für DIV 101
v2.5
\@@BCOR: Einzelzeichenscanner durch Mustererkennung ersetzt 77
\@@DIV: Einzelzeichenscanner durch Mustererkennung ersetzt 77
\datengerman: ngerman neu und identisch mit german 48
\datenorsk: Sprachauswahl um ngermen erweitert 55
\isopaper: neu 78
\typearea: „arithmetic overflow“ bei landscape behoben 104
\x@@paper: neue Mustererkennung für die Papiergröße . . 78
v2.5a
\isopaper: Ausgabe der gewählten Papiergröße entfernt 79
\ta@modiv: \ifdim statt \ifnum 96
\typearea: \ifdim statt \ifnum 101, 103
\ifdim statt \ifnum. 103
v2.5b
\datenorsk: Reaktivierung der Sprache geschieht nun via \languagename, soweit dies möglich ist 55
\typearea: bei der Mindestrandbetrachtung headinclude und footinclude berücksichtigen 103
v2.5c
\l@part: Formatierung bei Zeilenumbruch im part-Eintrag korrigiert 188
v2.5d
\addsec: \@mkboth bedingt durch \markright ersetzt 181
\closing: Abstand zwischen Schlussgruß und Signatur variabel 278
\presig@skip: Neues Makro, das den Abstand zwischen Schlussgruß und Signatur beinhaltet 278
\setpresigskip: Neues Makro, das den Abstand zwischen Schlussgruß und Signatur setzt 278
v2.5e
\@part: \numberline verwendet, damit die reservierte Part-Nummern-Breite in \l@part auch wirklich verwendet wird 171
\datenorsk: Workaround für Sprache nohyphenation durch Format mit Babel-Erweiterung aber kein Babel package geladen 55
\l@part: beinhaltet \sectfont eine Größenangabe, so wird

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

diese überschrieben 188
geänderte part-Formatierung ragte in den linken Rand . 188
v2.5f
\partformat: überflüssiges Leerzeichen am Ende entfernt 169
v2.5g
\@makecaption: \@makecaption per \newcommand definiert statt mit \long\def 118
neben den komplett hängenden Titeln sind nun auch nicht hängende oder teilweise hängende möglich 116
\labelitemi: Mathemodus eliminiert 214
\labelitemii: \textendash verwendet 214
\labelitemiii: Mathemodus eliminiert 214
\labelitemiv: Mathemodus eliminiert 214
v2.5h
\frontmatter: Option twoside beachten 166
\mainmatter: Option twoside beachten 166
\typearea: Text der Warnung korrigiert 101
v2.6
\@pt@@scan: neu (intern) 59
\@pt@scan: neu (intern) 59
General: Einlesen der Schriftgrößen-Options-Datei funktioniert nun mit diversen Größen 60
\areaset: \headheight von \f@size statt von \@ptsize abhängig gemacht 106
\cc: \def durch \newcommand* ersetzt 279
\encl: \def durch \newcommand* ersetzt 279
\SetDIVList: \ta@divlist enthält Werte für Schriftgrößen ab 1pt 76
\SetXDIVList: neu 76
\ta@divfor: Berücksichtigung der erweiterten Liste 98
\ta@divlist: die Liste umfasst nun Werte ab 1 pt statt ab 10 pt. 92
\ta@divlistelse: neu (intern) 92
v2.6a
\@spart: zum Löschen der Kolumnentitel wird nun \@mkboth verwendet 171
v2.6b
bibtotoc: \@mkboth eingefügt 206
bibtotocnumbered: \@mkboth eingefügt 206
v2.6c
\@endpart: \@endpart fügt nur noch in beidseitigen Dokumenten bei Verwendung von openright eine Leerseite ein 171
\bib@heading: die Präambel wird gesetzt 205
\bibpreamble: neue (intern) 206
bibtotoc: \bibpreamble eingefügt 206
bibtotocnumbered: \bibpreamble eingefügt 206
\idx@@heading: im Index kann nach der Überschrift eine Präambel gesetzt werden . 210
\idx@heading: Verwendung von \index@preamble eingefügt 211
\index@preamble: neu (intern) 211
\setbibpreamble: neu 206
\setindexpreamble: neu . . . 211
v2.6d
\bibpreamble: Makro funktioniert nachdem \let durch \newcommand* ersetzt wurde 206
\setindexpreamble: Befehl funktioniert nachdem \let durch \newcommand* ersetzt wurde 211
v2.7
\autodot: neu 168
\chapterformat: einfaches Leerzeichen durch \enskip ersetzt 169
bibtotoc: Klammerung korrigiert 206

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

bibtotocnumbered: Klammerung korrigiert 206
\othersectionlevelsformat: \quad durch \enskip ersetzt 170
neu 170
v2.7a
\addsec: \addsec für eine bessere Unterstützung von hyperref geändert 181
abstract: statt \section* wird nun \addsec* verwendet, damit die Kolumnentitel korrekt behandelt werden . . . 159
\idx@@heading: für scrbook und scrreprt die Link-Erzeugung mit hyperref verbessert . 210
v2.7b
\@makechapterhead: Kapitel können grundsätzlich mit einer Präambel versehen werden 174
\@makeschapterhead: Kapitel können grundsätzlich mit einer Präambel versehen werden 178
\@topnewpage: neu (LaTeX intern) 174
\bib@heading: \setchapterpreamble wird verwendet 205
bibtotocnumbered: \setchapterpreamble wird verwendet 206
\if@at@twocolumn: neu (intern) 174
\scr@topnewpage: neu (intern) 174
\setchapterpreamble: neu . 187
v2.7c
headings: vertikale Abstände nach der Kapitelüberschrift geringfügig verändert und mit Leim versehen, um mit \flushbottom zu besseren Ergebnissen zu gelangen . 163
v2.7e
\@saddchap: fehlende, schließende Klammer ergänzt 181
v2.7f
\@saddchap: fehlendes Klammerpaar ergänzt 181
v2.7g
\@@headlines: neue Mustererkennung für die Anzahl der Kopfzeilen 77
\@topnewpage: vergessenes \long ergänzt 174
pagesize: neue Option 84
\areaset: auch hier existiert der symbolische Wert current für BCOR 106
pdftex: neue Option 88
\ifpdfoutput: neu 23
\ta@bcor: \newlength and Stelle von \newdimen für die Definition verwendet 76
\typearea: die neuen Werte current, default, calc, classic werden verstanden 99
dvips: neue Option 88
v2.7h
\partformat: verlorenen Backslash in \autodot wieder eingefügt 169
v2.7i
bibtotocnumbered: \chapter statt \addchap 206
\othersectionlevelsformat: \let durch \def ersetzt . 170
\ta@temp@goodwidth: Berechnung ist robuster gegen arithmetischen Überlauf bei sehr großen Schriften 96
v2.7j
\bib@heading: Setzen der Präambel in die thebibliography-Umgebung verschoben . . . 205
\bibpreamble: Setzen der Präambel in die thebibliography-Umgebung verschoben . . . 206
theindex: Setzen der Präambel in die theindex-Umgebung verschoben 211
bibtotoc: Setzen der Präambel in die thebibliography-Umgebung verschoben . . . 206
bibtotocnumbered: Setzen der Präambel in die thebibliography-Umgebung verschoben 206

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\idx@@heading: Setzen der Präambel in die theindex-Umgebung verschoben . . . 210
\idx@heading: Setzen der Präambel in die theindex-Umgebung verschoben . . . 211
\setbibpreamble: \renewcommand durch \def ersetzt zur Erhöhung der natbib-Kompatibilität 206

v2.8
\appendix: \appendixmore wird beachtet 166
\chapapp: neu 170
\chapappifprefix: neu 170
\chapterformat: \chapappifprefix eingefügt 169
\chaptermarkformat: \chapappifprefix eingefügt 169
chapterprefix: neue Option 162
noappendixprefix: neue Option 162
nochapterprefix: neue Option 162
appendixprefix: neue Option 162

v2.8a
\cleardoubleemptypage: neu 136
\cleardoubleplainpage: neu 136
\cleardoublestandardpage: neu 136
cleardoubleempty: neue Option 136
cleardoubleplain: neue Option 136
cleardoublestandard: neue Option 136

v2.8b
\float@listhead: neu (für alle Paketautoren) 200
listotoc: Verwendung von \float@headings 196

v2.8c
\@addpart: neu (intern) 173
\@saddpart: neu (intern) . . . 173
\addpart: neu 173
\capfont: \normalcolor eingefügt 115
\caplabelfont: \normalcolor eingefügt 116
\headfont: \normalcolor eingefügt 138
\pnumfont: \normalcolor eingefügt 138
\sectfont: \normalcolor eingefügt 183

v2.8d
\addchap: \chapterpagestyle statt plain 179
\addpart: \partpagestyle statt plain 173
\chapter: \chapterpagestyle statt plain 174
\chapterpagestyle: neu . . . 142
theindex: \indexpagestyle statt plain 211
\indexpagestyle: neu 142
\maketitle: \titlepagestyle statt plain 155
\part: \partpagestyle statt plain 171
\partpagestyle: neu 142
\titlepagestyle: neu 142

v2.8e
\@@maybeautodot: neu (intern) 168
\@chapter: \@maybeautodot wird aufgerufen 174
\@maybeautodot: neu (intern) 167
\@part: \@maybeautodot wird aufgerufen 171
General: \@maybeasf entfernt 170
\@thepart entfernt 170
\appendix: \@altsecnumformattrue wird nicht länger ausgeführt 166
\if@autodot: neuer Schalter 167

v2.8f
\setpartpreamble: neu 187

v2.8g
\@chapter: per \float@addtolists wird nun auch ein vertikaler Abstand in Listen des float-Pakets eingebaut 174
\float@listhead: \float@headings umbenannt in \float@listhead 200
liststotoc: \float@headings umbenannt in \float@listhead 196

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

v2.8h
- \@makecaption: Unterscheidung zwischen Über- und Unterschriften 118
- table: bei Tabellen kann der Default für \caption von Unter- auf Überschriften geändert werden 127
- tablecaptionabove: neue Option 111
- tablecaptionbelow: neue Option 111
- \captionabove: neu 112
- \captionbelow: neu 112
- textsubscript: neu 44
- \if@captionabove: neu 112
- \if@tablecaptionabove: neu Option 109
- figure: \caption wird explizit zu \captionbelow 126
- \scr@caption: neu (intern) . 112
- @textsubscript: neu (intern) 44

v2.8i
- General: \baselinestretch wird nicht umdefiniert . . . 147
- halfparskip: neue Option . . 145
- halfparskip*: neue Option . 145
- halfparskip+: neue Option . 145
- parindent: neue Option 145
- parskip: neue Option 145
- parskip*: neue Option 145
- parskip+: neue Option 145
- \f@parfillskip: neu (intern) 144
- \f@parskip: neu (intern) . . . 144
- \scr@parfillskip: neu (intern) 145
- \scr@parskip: neu (intern) . 145

v2.8j
- \@makecaption: \@makecaption verwendet \@@makecaption 118
- \floatc@komascript: neu (intern) 124
- \fs@komaabove: neu (intern) 124
- \fs@komabelow: neu (intern) 124
- \if@capbreak: neu 116
- \LT@makecaption: neu (intern) 123
- \setcapindent: \if@capbreak wird verwendet 116

v2.8k
- tablecaptionabove: aus typearea in die Klassen verschoben 111
- tablecaptionbelow: aus typearea in die Klassen verschoben 111
- \if@tablecaptionabove: aus typearea in die Klassen verschoben 109
- \LT@makecaption: keine Umdefinierung, wenn das caption2-Paket erkannt oder die Option origlongtable gesetzt wurde 123

v2.8l
- \@addpart: erst eintragen, dann ausgeben 173
- halfparskip-: neue Option . 145
- parskip-: neue Option 145
- \if@komalongtable: neue Option 111
- \listoffigures: Gruppe eingefügt und \parskip auf 0 gesetzt 201
- \listoftables: Gruppe eingefügt und \parskip auf 0 gesetzt 201
- origlongtable: neue Option 111
- \setcapindent: \edef\cap@indent statt \def\cap@indent . . 116
- \tableofcontents: Gruppe eingefügt und \parskip auf 0 gesetzt 200

v2.8n
- \@beginletterhook: Neu . . . 277
- \@firstheadfootfield: Neu 258
- \@foldmark: Neu 255
- \@savevbox: Neu 257
- \addrchar: Neu 316
- \addrentry: Neu 316
- \AtBeginLetter: Neu 277
- \cc: Komplett überarbeitet . 279
- refline: Neue Option 241
- \encl: Komplett überarbeiten 279
- \if@refwide: Neuer Schalter 241
- \move@frompl: Neu 254
- \move@frompt: Neu 254
- \move@topl: Neu 254

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\move@topt: Neu 254
\opening: Komplett überarbeitet 277
\ps: Komplett überarbeitet . 279
\vb@t@z: Neu 255
v2.8o
\@datefield: für den Brieftyp KOMAold anschließend einen zusätzlichen vertikalen Abstand eingefügt 270
\@locfield: \noindent eingefügt 265
pagefoot: neues Element pagefoot 138
pagehead: neues Element pagehead 138
pagenumber: neues Element pagenumber 138
pagination: neues Alias-Element 138
\appendix: \par eingefügt . . 166
\chapappifchapterprefix: Argument ist nicht optional . 170
neu 170
\chapappifprefix: obsolet . 170
\chapterformat: \chapappifchapterprefix statt \chapappifprefix . 169
\chaptermarkformat: \chapappifchapterprefix statt \chapappifprefix . 169
minisec: neuer Ersatz für das Element minisec 184
enlargefirstpage: Neue Option 243
\scr@fnt@caption: neues Element caption 116
\scr@fnt@captionlabel: neues Element captionlabel . . 116
\scr@fnt@chapter: neues Element chapter 184
\scr@fnt@descriptionlabel: neues Element descriptionlabel 217
\scr@fnt@paragraph: neues Element paragraph 184
\scr@fnt@section: neues Element section 184
\scr@fnt@sectioning: neues Element sectioning 184
\scr@fnt@subparagraph: neues Element subparagraph . . 184
\scr@fnt@subsection: neues Element subsection 184
\scr@fnt@subsubsection: neues Element subsubsection 184
\scr@fnt@title: neues Element title 158
\scr@fnt@wrn@pagehead: neue Warnung für Element pagehead 138
\setkomafont: neu 72
\size@chapter: neu (intern) 162
\size@paragraph: neu (intern) 162
\size@part: neu (intern) . . . 162
\size@partnumber: neu (intern) 162
\size@section: neu (intern) 162
\size@subparagraph: neu (intern) 162
\size@subsection: neu (intern) 162
\size@subsubsection: neu (intern) 162
v2.8p
\@@makechapterhead: neu (intern) 174
\@@makeschapterhead: neu (intern) 178
\@endpart: \use@preamble ersetzt \@part@preamble . . 171
\vbox eingefügt 171
\@makechapterhead: es gibt verschiedene Arten von Präambeln 174
\@maketitle: \titlefont an Stelle von \sectfont . . . 272
\@part: \sectfont wird nun vor \size@part und \size@partnumber aufgerufen 171
\addtokomafont: neu 73
\chapter@o@preamble: neu (intern) 187
\chapter@u@preamble: Ersatz für \chapter@preamble . 187
neu (intern) 187
\l@addto@macro: neu (intern) 30

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\maketitle: \titlefont an Stelle von \sectfont . . . 154
\paragraph: \sectfont wird nun vor \size@paragraph aufgerufen 181
\part: Präambel über der Überschrift hinzugefügt . . 171
\part@preamble@o: neu (intern) 187
\part@preamble@u: Ersatz für \part@preamble 187
neu (intern) 187
\scr@fnt@part: neues Element part 184
\scr@fnt@partnumber: neues Element partnumber 184
\section: \sectfont wird nun vor \size@section aufgerufen 181
\set@@@@preamble: neu (intern) 184
\set@@@preamble: neu (intern) 184
\set@@preamble: neu (intern) 184
\set@preamble: neu (intern) 184
\setchapterpreamble: Verwendung von \set@preamble 187
\setpartpreamble: Verwendung von \use@preamble 187
\subparagraph: \sectfont wird nun vor \size@subparagraph aufgerufen 181
\subsection: \sectfont wird nun vor \size@subsection aufgerufen 181
\subsubsection: \sectfont wird nun vor \size@subsubsection aufgerufen 181
\titlefont: neu 158
\use@preamble: neu (intern) 184
\usekomafont: neu 74
v2.8q
\@@@@captionbeside: neu (intern) 113
\@@@captionbeside: neu (intern) 113
\@@PapersizeWarning: Neu (intern) 291
\@@captionbeside: neu (intern) 113
\@@makecaption: Problem mit doppelten Fußnoten behoben 121
\@@makechapterhead: \raggedsection wird bei Option chapterprefix auch auf den Präfix angewendet 174
\@@setcapmargin: neu (intern) 120
\@@setkomavar: Neu 283
\@@xtrimtempa: Neu (intern) 278
\@PapersizeWarning: Neu (intern) 291
\@addmargin: neu (intern) . . 216
\@addrfield: \parfillskip lokal geändert 260
Sonderbehandlung für Adressfeld rechts 260
\@addtoplength: Neu 288
\@alignfrom: Neu 235
\@captionbeside: neu (intern) 113
\@datefield: \@parskipfalse\@parskip@indent 270
\rlap eingefügt 270
Leertests entfernt, da in \@set@reffield bereits enthalten 270
Schalter \if@tempwa wird bereits hier auf true gesetzt 269
\@emailfalse: Neu 237
\@emailtrue: Neu 237
\@faxfalse: Neu 236
\@faxtrue: Neu 236
\@firstheadfootfield: \parfillskip lokal auf 0 gesetzt 258
Fuß explizit in eine Box gesetzt 258
\@gen@firsthead: Neu 245
\@komavar@err: Neu 284
\@l@number: neu (intern) . . . 198
\@list@extra: neu (intern) . 150
\@locfield: \noindent nach dem Logo 265
Berechnung der Breite hierher verschoben 265
Sonderbehandlung für Adressfeld rechts 265
\@logofalse: Neu 237
\@logotrue: Neu 237

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\@makecaption: die Sonderbehandlung für einzeilige *captions* ist abschaltbar 118
Titel können in ihrer Breite beschränkt werden 118
\@maketitle: \@parskipfalse\@parskip@indent 272
\@newplength: Kompatibilität zu scrlettr.cls aufgegeben 287
Neu 287
\@ogobble: Neu 265
\@pageat: Neu 243
\@part: \@parskipfalse\@parskip@indent 171
\@phonefalse: Neu 236
\@phonetrue: Neu 236
\@rulefrom: Neu 236
\@saddchap: \AfterPackage zur Überprüfung der hyperref-Version verwendet 181
\@savevbox: zusätzlich ein optionales Argument 257
\@savevbox@warningfalse: Neu 257
\@set@reffields: Neu 270
\@setcapmargin: neu (intern) 120
\@setif: Neu 234
\@setkomaname: Neu 283
\@setkomavar: Neu 283
\@setplength: Neu 288
\@spart: \@parskipfalse\@parskip@indent 171
\@subjfield: \@parskipfalse\@parskip@indent 272
\centerline durch \centering ersetzt 272
wird mit hängendem Einzug gesetzt 272
\@tempdimaismaxwidthof: Neu (intern) 278
\@typearea: Textbreite berücksichtigt Option mpinclude 93
\@usekomaname: Neu 284
\@usekomavar: Neu 284
\@wwwfalse: Neu 237
\@wwwtrue: Neu 237
\@xtrimtempa: Neu (intern) . 278
General: Initialisierung mit Hilfe der symbolischen Werte current 107
backaddress: Neu 238, 280
backaddressseparator: Neu 280
\addrcolengthplength: Neu . 288
\adrchar: \adrchar verwendet das neue \addrchar 316
\adrentry: \adrentry verwendet das neue \addrentry 316
\after@starttoc: neu (intern) 199
pagenumber: Neu 243
captionbeside: neu 112
date: Neu 281
faxseparator: Neu 280
\before@starttoc: neu (intern) 198
\bibpreamble: wieder \let . 206
\c@pti@nb@sid@: neu (intern) 114
\c@pti@nb@sid@b@x: neu (intern) 114
\cap@left: neu (intern) 118
\cap@margin: neu (intern) . . 120
\cap@right: neu (intern) . . . 118
\cap@width: neu (intern) . . . 118
\closing: \@parskipfalse\@parskip@indent 278
darf \\[...] enthalten . . . 278
nicht mehr \long 278
ccseparator: Neu 280
\datecroatian: Neu 48
\datedutch: Neu 48
addmargin: neue Umgebung . 216
addmargin*: neue Umgebung 216
addrfield: Neu 237
\defaultreffields: Neu . . . 270
\dictum: neu 41
\dictumauthorformat: neu . . 41
\dictumwidth: neu 41
\encl: Fehlendes % ergänzt . 279
letter: Fußnotenzähler wird zurückgesetzt 276
optionales Argument hinzugefügt 276
\firsthead: \@alignfrom auf 10000 gesetzt 245
\float@listhead: dynamische Anpassung an die Nummer 200

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\footnoterule: \raggedbottom-Verbesserung eingefügt . . 130
\ftn@font: neu (intern) 132
\ftnm@font: neu (intern) . . . 132
\ftntm@font: neu (intern) . . 133
thebibliography: Erkennung und gesonderte Behandlung leerer Präambeln 207
phoneseparator: Neu 280
dictum: neues Element 41
dictumauthor: neues Element 41
dictumtext: neues Element . . 41
\if@dynlist: neuer Schalter 193
\if@email: Neu 237
\if@fax: Neu 236
\if@firstfoot: Neuer Schalter (intern) 243
\if@firsthead: Neuer Schalter (intern) 235
\if@komavarempty: Neu 285
\if@logo: Neu 237
\if@phone: Neu 236
\if@tocleft: neuer Schalter 192
\if@www: Neu 237
\ifkomavarempty: Neu 285
\ifonelinecaptions: neuer Schalter 109
\ifpdfoutput: \pdfoutput wird nicht mehr zu \relax, wenn zuvor undefiniert 23
darf bereits definiert sein . . . 23
Verwendung von \scr@ifpdfoutput 23
\ifscap@alt: neuer Schalter 113
\ifthispageodd: neu 42
signature: Neu 280
\is@thispageodd: neu (intern) 43
listsindent: neue Option . . 196
listsleft: neue Option 196
liststotocnumbered: dem langjährigen Druck nachgegeben 196
title: Neu 280
place: Neu 280
placeseparator: Neu 280
cleardoublestandard: \cleardoublestandard@page statt \cleardoublestandard 136
\LetterOptionNeedsPapersize: Neu 291
\listoffigures: \@parskipfalse \@parskip@indent 201
\listoftables: \@parskipfalse \@parskip@indent 201
\LoadLetterOption: Neu . . . 290
\LOPNP@option: Neu (intern) 291
\LOPNP@size: Neu (intern) . . 291
\LT@array: neu (intern) 123
\LT@caption: neu (intern) . . 123
\LT@makecaption: Breite und Anordnung wird über KOMA-Script-Makros bestimmt 123
Verwendung von \AfterPackage 123
\ltr@len@@firstfootvpos: Neu 253
\ltr@len@backaddrheight: Neu 251
\ltr@len@bfoldmarkvpos: Neu 251
\ltr@len@firstfootwidth: Neu 253
\ltr@len@firstheadvspos: Neu 253
\ltr@len@firstheadwidth: Neu 253
\ltr@len@foldmarkhpos: Neu 251
\ltr@len@fromrulewidth: Neu 253
\ltr@len@locwidth: Neu . . . 252
\ltr@len@refaftervskip: Neu 252
\ltr@len@refvpos: Neu 252
\ltr@len@refwidth: Neu . . . 252
\ltr@len@sigbeforevskip: Neu 253
\ltr@len@sigindent: Neu . . 253
\ltr@len@specialmailindent: Neu 251
\ltr@len@specialmailrightindent: Neu 251
\ltr@len@tfoldmarkvpos: Neu 251
\ltr@len@toaddrheight: Neu 252
\ltr@len@toaddrhpos: Neu . 251
\ltr@len@toaddrindent: Neu 252
\ltr@len@toaddrvpos: Neu . 251
\ltr@len@toaddrwidth: Neu 252
emailseparator: Neu 280

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\marginline: fehlende Klammern ergänzt 111
\minisec: \nobreak nach \sectfont behebt einen Bug im color Paket 182
enclseparator: Neu 280
onelinecaption: neue Option 111
\new@tpo@label: neu (intern) 42
\newcaptionname: neu 45
\newkomavar: Neu 282
invoice: Neu 281
toaddress: Neu 281
location: Neu 280
locfield: Neu 239
tocindent: neue Option 193
tocleft: neue Option 193
foldmarks: Neu 240
toname: Neu 281
noonelinecaption: neue Option 111
\opening: \@PapersizeWarning 277
\@parskipfalse\@parskip@indent 277
yourmail: Neu 281
yourref: Neu 281
\paragraph: \nobreak nach \sectfont behebt einen Bug im color Paket 181
\part: \nobreak nach \sectfont behebt einen Bug im color Paket 171
specialmail: Neu 280
mpexclude: neue Option 82
mpinclude: neue Option 82
\presig@skip: Ersetzt durch Pseudo-Länge sigbeforevskip 278
\providecaptionname: neu . . 45
\raggeddictum: neu 41
\raggeddictumauthor: neu . . 41
\raggeddictumtext: neu 41
\removereffields: Neu 270
\renewcaptionname: neu 45
urlseparator: Neu 280
fromaddress: Neu 281
fromalign: Neu 235
frombank: Neu 281
fromemail: Neu 236, 281
fromfax: Neu 236, 281
fromlogo: Neu 237, 280
fromname: Neu 281
fromphone: Neu 236, 281
fromrule: Neu 236
fromurl: Neu 237, 281
\scap@caption: neu (intern) 112
\scap@currentlabel: neu (intern) 112
\scap@move: neu (intern) . . . 112
\scap@pos: neu (intern) 109
\scap@valign: neu (intern) . 109
\scap@width: neu (intern) . . 112
\scr@fnt@footnote: neues Element footnote 132
\scr@fnt@footnotenlabel: neues Element footnotelabel 132
\scr@fnt@footnotenumber: neues Element footnotenumber 132
\scr@fnt@footnoteref: neues Element footnoteref . . . 133
\scr@fnt@footnotereference: neues Element footnotereference 133
\scr@fnt@instead@footnotetext: neuer Ersatz für Element footnotetext 132
\scr@fnt@subject: neues Element subject 158
\scr@ifpdfoutput: funktioniert 23
neu (intern) 23
\scr@topnewpage: fehlende Klammern ergänzt 174
\scr@tpo: neu (intern) 42
\section: \nobreak nach \sectfont behebt einen Bug im color Paket 181
\set@@@@preamble: fehlende Klammern ergänzt 184
\set@l@number: neu (intern) 198
\set@preamble@hpos: neu (intern) 186
\set@preamble@vpos: neu (intern) 187
\set@tempdima@hw: neu (intern) 138

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\setcaphanging: wird nach dem Laden von caption2 umdefiniert 117
\setcapindent: wird nach dem Laden von caption2 umdefiniert 116
\setcapmargin: neu 120
\setcapwidth: neu 118
\setkomavar: Neu 283
\setlengthtoplength: Neu . 288
\setpresigskip: Mit Hilfe der neuen Pseudo-Längen definiert 278
\subject@font: neu (intern) 158
\subparagraph: \nobreak nach \sectfont behebt einen Bug im color Paket 181
\subsection: \nobreak nach \sectfont behebt einen Bug im color Paket 181
\subsubsection: \nobreak nach \sectfont behebt einen Bug im color Paket 181
\ta@divfor: Berücksichtigung der Option 'mpinclude' . . . 98
\ta@temp@goodwidth: \normalfont \normalsize hinzugefügt . 96
\tableofcontents: \@parskipfalse \@parskip@indent 200
\if@tocleft bearbeiten . . 200
\toc@l@number: neu 200
\typearea: DIVcalc liefert im zweispaltigen Satz (wie dokumentiert) breitere Spalten 102
Berüchsichtigung von Option mpinclude bei der *DIV*-Berechnung 103
der Wert last wird verstanden 99
subject: Neu 242, 281
subjectseparator: Neu 280
numericaldate: Neue Option (an Stelle von scrdate und orgdate) 45
\usekomavar: Neu 284
\useplength: Neu 287
customer: Neu 281
myref: Neu 281
v2.95
\@@makecaption: Korrekturen für vertikalen Abstand und *float*-Paket 121
\@CheckKOMAScriptVersion: neu (intern) 10
\@addmargin: \item muss keine „[]“-Klammern mehr verarbeiten 216
\@addpart: kann Kolumnentitel erzeugen 173
\@datefield: Absatzeinzug mit \setparsizes abschalten 270
\@foldmark: \vskip in \vb@t@z verschoben, damit es vom Satzspiegel unabhängig wird 255
\@maketitle: \usekomafont and Stelle fon \titlefont 272
\@part: \@parskipfalse und 171
\@parskipfalse und \@parskip@indent ersetzt 171
\centering und \@parskipfalse \@parskip@indent vertauscht 171
\@setif: Nur vor Version 2.95 234
\@spart: \@parskipfalse und \@parskip@indent ersetzt 171
\centering und \@parskipfalse\@parskip@inden vertauscht 171
\@subjfield: Absatzeinzug mit cssetparsizes einstellen . . . 272
\@typearea: \selectfont eingefügt 93
General: erste Version aus der Aufteilung von scrclass.dtx 232, 235, 282, 287, 290, 316
erste Version aus der Aufteilung von scrclass.dtx . . . 39, 45, 59, 76, 109, 129, 135, 144, 152, 160, 192, 203, 210, 213
First version split from hugh scrclass.dtx 218
Beim Laden von typearea wird \KOMAScriptVersion verwendet 93
erste Version aus der Aufteilung von scrclass.dtx . . . 10
es wird erst unmittelbar vor dem Schließen der Haupt-

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

aux-Datei in diese geschrieben 168
backaddress: jetzt mit \newkomafont 250
pagesize: neu implementiert . 84
halfparskip: obsolet 145
halfparskip*: obsolet 145
halfparskip+: obsolet 145
halfparskip-: obsolet 145
\aliaskomafont: neu 75
paper: neu implementiert 89
\appendix: \newcommand* durch \gdef ersetzt 166
Ausführung von \appendixmore vereinfacht 166
\areaset: am Ende von \areaset wird noch \@typeare@end ausgeführt 106
parindent: obsolet 145
parskip: primäre keyval-Option 145
parskip*: obsolet 145
parskip+: obsolet 145
parskip-: obsolet 145
\AtEndLetter: Neu 277
\BeforeClosingMainAux: new 228
\ClassName: Neu 10
\cleardoubleusingstyle: neu 136
\closing: Absatzformatierung mit \setparsizes einstellen 278
BCOR: als keyval-Option implementiert 82
\default@par@update: neues internes Macro 145
\deffootnote: Absatzeinzug korrigiert 132
pdftex: neu implementiert . . . 88
headexclude: Verwendung von scrkeyvl 82
headinclude: Verwendung von scrkeyvl 82
headlines: mit keyval neu implementiert 81
version: Option kann nur beim Laden der Klasse gesetzt werden 232
\f@parfillskip: Bedeutung geändert 144
\f@parindent: neu (intern) . 144
\f@parskip: Bedeutung geändert 144
\footincludeoff: ersatzlos gestrichen 137
\footincludeon: ersatzlos gestrichen 137
\footnoterule: .05fil statt .005fil 130
\footseplineoff: ersatzlos gestrichen 137
\footseplineon: ersatzlos gestrichen 137
\headincludeoff: ersatzlos gestrichen 137
\headincludeon: ersatzlos gestrichen 137
\headseplineoff: ersatzlos gestrichen 137
\headseplineon: ersatzlos gestrichen 137
\if@atdocument: neuer Schalter 18
\if@fincl: Verwendung von scrkeyvl 82
\if@hincl: Verwendung von scrkeyvl 82
\if@mincl: Verwendung von scrkeyvl 82
\ifdvioutput: neu 24
\ifnotundefined: neu 20
\ifpdftex: neu 23
\ifpsoutput: neu 24
\ifstr: if- und then-Teil werden nicht gefressen 20
\ifundefinedorrelax: neu . . 20
\ifVTeX: neu 23
final: obsolete Option 39
\iso@@paper: neu (intern) . . . 80
\iso@paper: neu (intern) 80
\isopaper: es sind auch Großbuchstaben erlaubt 78
title: jetzt mit \newkomafont 250
DIV: als keyval-Option implementiert 82
DIVcalc: über Option DIV reimplementiert 83
DIVclassic: über Option DIV reimplementiert 83
\KOMAClassName: Neu 10
\KOMAScriptVersion: neu . . . 10

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\l@addto@macro: korrigiert, damit es sich bezüglich Erweiterung um Makrodefinitonen mit Argumenten wie \g@addto@macro verhält . . 30
\lco@test: Neu 290
\LCOWarning: Neu 293
\LCOWarningNoLine: Neu . . . 293
cleardoublepage: neuer Schalter 136
\let@as@internal@defined: neu (intern) 18
\listoffigures: \@parskipfalse und \@parskip@indent ersetzt 201
\listoftables: \@parskipfalse und \@parskip@indent ersetzt 201
\maybe@pair: neu (intern) . . . 91
onecolumn: Option in das typearea-Paket verlagert 84
\newcaptionname: Ermittlung der aktuellen Sprache funktioniert auch, wenn \languagename mit seltsamen catcodes erstellt wurde. 45
\newkomafont: neu 75
footexclude: Verwendung von scrkeyvl 82
footinclude: Verwendung von scrkeyvl 82
\opening: Absatzformatierung mit \setparsizes einstellen 277
\pagemark: nun für alle Klassen 138
\par@update: neues internes Macro 145
\par@updaterelative: neues internes Macro 148
\parfillskip@update: neues internes Macro 149
\parindent@update: neues internes Macro 149
\parskip@update: neues internes Macro 149
mpexclude: Verwendung von scrkeyvl 82
mpinclude: Verwendung von scrkeyvl 82
\providecaptionname: Ermittlung der aktuellen Sprache funktioniert auch, wenn \languagename mit seltsamen catcodes erstellt wurde. 45
draft: mit *keyval* reimplementiert 39
\recalctypearea: Neu 106
\renewcaptionname: Ermittlung der aktuellen Sprache funktioniert auch, wenn \languagename mit seltsamen catcodes erstellt wurde. 45
fromaddress: jetzt mit \newkomafont 249
fromname: jetzt mit \newkomafont 249
\scr@@@defaultunits: neues internes Macro 149
\scr@@defaultunits: neues internes Macro 149
\scr@defaultunits: neues internes Macro 149
\scr@fnt@subject: jetzt auch bei scrbook, scrreprt und scrartcl 158
\scr@hook@bcma: new (internal) 230
\scr@ifdvioutput: Expandierbarkeit verbessert 24
neu (intern) 24
\scr@ifpdfoutput: berücksichtigt VTEX 23
Expandierbarkeit verbessert 23
\scr@ifpdftex: neu (intern) . 23
\scr@ifpsoutput: Expandierbarkeit verbessert 24
neu (intern) 24
\scr@ifVTeX: neu 23
\scr@new@selectfont: neue Erweiterung 148
\scr@parfillskip: Bedeutung geändert 145
\scr@parindent: neu (intern) 145
\scr@parskip: Bedeutung geändert 145
\scr@pti@nerr@r: \OptionNotUsed statt \PackageError 81

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\scr@saved@checkend: new (internal) 229
\scr@selectfont: neue Erweiterung 148
\scr@v@2.95: Neues Macro . 233
\selectfont: neue Änderung 147
\set@dvipspagesize: neu (intern) 85
\set@pdftexpagesize: neu (intern) 85
\set@vtexpagesize: neu (intern) 85
\setindexpreamble: \let und \def 211
\setnot@dvipspagesize: neu (intern) 85
\setparsizes: neues Macro . 144
\subject@font: jetzt auch bei scrbook, scrreprt und scrartcl 158
\tableofcontents: \@parskipfalse und \@parskip@indent ersetzt 200
article beachtet ebenfalls tocleft 200
\toc@heading: auch für scrreprt und scrbook 199
\typearea: Warnung im Zweispaltenmodus korrigiert . . 104
subject: jetzt mit \newkomafont 250
\vb@t@z: Absatzformatierung mit \setparsizes 255
dvips: neu implementiert 88
twocolumn: Option in das typearea-Paket verlagert 84

v2.95a
\@@makecaption: diverse Korrekturen für den vertikalen Abstand u. a. bei Verwendung des *color*-Pakets . . . 121
\@addmargin: \@tmpcnta bleibt unverändert 216
\addpart: support of preamble added 173
\before@starttoc: Initialisierung im \@empty-Fall . . . 198
addmargin: \@listdepth korrigiert 216
addmargin*: \@listdepth korrigiert 216
\partheadendvskip: neue Konfiguratonsmöglichkeit . . . 171
\partheadmidvskip: neue Konfiguratonsmöglichkeit . . . 171
\partheadstartvskip: neue Konfiguratonsmöglichkeit 171
\raggedpart: neue Konfigurationsmöglichkeit 183
\ta@temp@goodwidth: Berechnung für große Schriften korrigiert 96
\typearea: Grundschriftgröße einstellen 100

v2.95b
\is@thispageodd: basiert nun auf \@newl@bel 43
\new@tpo@label: verwendet jetzt \@newl@bel 42
\scr@fnt@disposition: neues Element disposition . . . 184
\scr@fnt@sectioning: sectioning ist nun ein Alias für disposition 184

v2.95c
\@list@extra: \parsep fixed 150
\@maketitle: Absatzgrundeinstellungen 157
abstract: neue keyval-Option 153
abstractoff: Option ist obsolet 153
abstracton: Option ist obsolet 153
headsepline: BugFix: Argument bei \PassOptionsToPackage vertauscht 135
chapterprefix: als keyval-Option 162
\ifnotundefined: nutzt ε-TEX wenn möglich 20
\ifundefinedorrelax: über interne Anweisung definiert . 20
titlepage: als keyval-Option 152
\let@as@internal@defined: Warnung bei Umdefinierung 18
\maketitle: Absatzgrundeinstellungen 154
noappendixprefix: obsolete Option 162

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

nochapterprefix: obsolete Option 162
footsepline: BugFix: Argument bei \PassOptionsToPackage vertauscht 135
notitlepage: Option ist obsolet 152
appendixprefix: als keyval-Option 162
\scr@ifundefinedorrelax: neu (intern) 13
v2.96
\@addrfield: Benutzung von \backaddr@format 260
\@fontsizefilebase: neu (intern) 59
\@globaloptionslist: Klassenoptionen durchsuchen 15
new 15
\@loadwithoptions: new redefinition 223
\@onefilewithoption: new redefinition 224
\@ptsize: \@ptsize hat temporär eine andere Bedeutung 59
\@replacefilewithext: new helper command 224
\@typearea: \topskip wird gesetzt 96
General: Optionen werden abgearbeitet 18
Optionen werden in Reihenfolge der Angabe statt in Reihenfolge der Deklaration abgearbeitet. 92
\areaset: \topskip wird gesetzt 106
\backaddr@format: neu (intern) 238
\changefontsizes: neue Anweisung (benötigt ε-TEX) 65
\dont@let@as@internal@defined: neu (intern) 17
\InputIfFileExists: replacement code added 223
\KOMA@DeclareObsoleteOption: \CurrentOption wir geschützt 37
\KOMAoptions: Neuer Hook nach den Optionen 29
\let@as@internal@defined: Herausreichen von internen Makros als Benutzermakros kann verhindert werden . . . 18
internalonly: neu 17
fontsize: Funktion komplett geändert 60
\ReplaceClass: new command 224
\ReplaceInput: new command 224
\ReplacePackage: new command 224
\scr@loadwithoptions: new helper macro 223
\scr@onefilewithoption: new helper macro 224
\scr@replacefile@msg: new helper command 225
\scr@sr@co: neu (intern) 38
\ta@divfor: Schriftgröße muss nicht in ganzen pt sein . . . 98
\typearea: Warnung, wenn \typearea innerhalb einer Gruppe verwendet wird . . . 99
\usekomafont: Benutzung eines Fontalias korrigiert 74
v2.96a
\@chapterlistsgap: neues Macro (intern) 196
\@typearea@end: wird wieder immer aber mit neuer Bedeutung definiert 97
General: es wird \immediate geschrieben 168
\activateareas: D 97
\AfterCalculatingTypearea: neu 97
\bprot@dottedtocline: neu (intern) 191
\chapterformat: \mbox eingefügt, um \caps zu ermöglichen 169
\cleardoublestandardpage: korrekter Ersatzwert gesetzt 136
refline: vertauschte Bedeutung korrigiert 241
chapteratlists: neue Option 196

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\if@at@twocolumn: Abstand nach der Präfixzeile nur, wenn eines Präfixzeile ausgegeben wurde 178
\if@chaptertolists: neuer Schalter 196
minisec: minisec ist ein eigenes Fontelement 182
Alias für das Element minisec entfernt, da es nun ein eigenes Element ist 184
\l@chapteratlist: neues Macro 189
cleardoublepage: Wert current sollte nun funktionieren 136
\minisec: alle nicht benötigten \nobreak entfernt 182
Standardwert für \parfillskip (abhängig von Option version) 182
\paragraph: alle nicht benötigten \nobreak entfernt 181
\scr@fnt@chapterprefix: neues Element chapterprefix 184
\scr@v@2.95a: Neues Macro 233
\scr@v@2.95b: Neues Macro 233
\scr@v@2.96: Neues Macro . 233
\scr@v@2.96a: Neues Macro 233
\section: alle nicht benötigten \nobreak entfernt 181
Standardwert für \parfillskip (abhängig von Option version) 181
\size@chapter: letztes Element darf ein Argument erwarten, wenn es keine Auswirkung auf den Zeilen abstand hat 162
\size@chapterprefix: neu (intern) 162
\size@part: letztes Element darf ein Argument erwarten, wenn es keine Auswirkung auf den Zeilen abstand hat 162
\size@partnumber: letztes Element darf ein Argument erwarten, wenn es keine Auswirkung auf den Zeilen abstand hat 162
\subparagraph: alle nicht benötigten \nobreak entfernt 181
\subsection: alle nicht benötigten \nobreak entfernt 181
Standardwert für \parfillskip (abhängig von Option version) 181
\subsubsection: alle nicht benötigten \nobreak entfernt 181
Standardwert für \parfillskip (abhängig von Option version) 181
v2.96b
\l@chapter: Umbruch zwischen \chapter-Eintrag im Inhaltsverzeichnis und übergeordneten Einträgen verhindern 188
Umbruch zwischen \chapter-Eintrag im Inhaltsverzeichnis und untergeordneten Einträgen verhindern 188
\l@paragraph: Umbruch zwischen \paragraph-Eintrag im Inhaltsverzeichnis und übergeordneten Einträgen verhindern 191
Umbruch zwischen \paragraph-Eintrag im Inhaltsverzeichnis und untergeordneten Einträgen verhindern 191
\l@part: Umbruch zwischen \part-Eintrag im Inhaltsverzeichnis und untergeordneten Einträgen verhindern . . . 188
\l@section: Umbruch zwischen \section-Eintrag im Inhaltsverzeichnis und übergeordneten Einträgen verhindern 189
Umbruch zwischen \section-Eintrag im Inhaltsverzeichnis und untergeordneten Einträgen verhindern 189

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\l@subparagraph: Umbruch zwischen \subparagraph-Eintrag im Inhaltsverzeichnis und übergeordneten Einträgen verhindern 191
Umbruch zwischen \subparagraph-Eintrag im Inhaltsverzeichnis und untergeordneten Einträgen verhindern 191
\l@subsection: Umbruch zwischen \subsection-Eintrag im Inhaltsverzeichnis und übergeordneten Einträgen verhindern 191
Umbruch zwischen \subsection-Eintrag im Inhaltsverzeichnis und untergeordneten Einträgen verhindern 191
\l@subsubsection: Umbruch zwischen \subsubsection-Eintrag im Inhaltsverzeichnis und übergeordneten Einträgen verhindern 191
Umbruch zwischen \subsubsection-Eintrag im Inhaltsverzeichnis und untergeordneten Einträgen verhindern 191
v2.97
\@seccntformat: Workaround für das blöde \protect . . 170
\othersectionlevelsformat: statt einem Argument jetzt drei 170
\scr@v@2.97: Neues Macro . 233
\setcapwidth: Mindestbreite für captions ist 0 pt 118
v2.97a
\scr@@loadwithoptions: new helper macro 223
\scr@v@2.97a: Neues Macro 233
v2.97b
\if@at@twocolumn: \endgraf korrigiert 178
\scr@ifundefinedorrelax: vermurkste Logik korrigiert . . 13
\scr@v@2.97b: Neues Macro 233
v2.97c
\@@makecaption: \scr@smashdp eingefügt 122
\unskip\strut eingefügt, um die vertikale Ausrichtung im *above*-Fall in eingen Fällen zu verbessern 121
\@datefield: Neue Font-Elemente 270
Neue Fontelemente 269
\@foldmark: Dicke konfigurierbar 255
Schrift (Farbe) konfigurierbar 255
\@gen@firsthead: Dicke der Linie konfigurierbar 245
Schrift (Farbe) der Linie konfigurierbar 245
\@pt@@scan: \KOMA@UseObsolete durch \KOMA@UseObsoleteOption ersetzt 59
\@subtitle: neu (intern) . . . 153
\@typearea: \maxdepth wird gesetzt 96
General: \lco@test bei den geparsten Optionen eingefügt 91
Option twoside bei den Voreinstellungen rausgenommen 40
Option twoside wird explizit passend gesetzt 92
Optionen werden mit \KOMAProcessOptions abgearbeitet 18
Optionen, die der Voreinstellung ohnehin entsprechen, werden nicht explizit gesetzt 40, 92
Verwendung von \KOMAExecuteOptions an Stelle von \KOMAoptions und \KOMAProcessOptions an Stelle von \ProcessOptions 40, 92
partentry: neues Font-Element 183
partentrypagenumber: neues Font-Element 183
\changefontsizes: \dblfloatsep setzen 70
\intextsep setzen 70
\partopsep setzen 70
\datenorsk: Text einer Warnung geändert 55
addressee: Neues Font-Element 250

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

headsepline: Option headinclude wird nicht mehr an typearea geschickt 135
sectionentry: neues Font-Element 183
sectionentrypagenumber: neues Font-Element 183
refname: Neues Font-Element 250
refvalue: Neues Font-Element 250
\flushbottom: neue Erweiterung 277
chapterentry: neues Font-Element 183
chapterentrypagenumber: neues Font-Element 183
\if@fincl: direkte Initialisierung über \if@fsl, falls definiert 82
\if@hincl: direkte Initialisierung über \if@hsl, falls definiert 82
\if@twoside: Schalter passend setzen 136
\ifscr@key@found@: Neuer Schalter (intern) 28
\ifundefinedorrelax: umbenannt von \ifundefined . . 20
\ifVTeX: umdefiniert von \ifvtex 23
\KOMA@key: neues optionales Argument 15
\KOMA@options: Neues Makro (intern) 29
\KOMA@UseObsoleteCommand: neu (intern) 38
\KOMA@UseObsoleteOption: neu (intern) 38
\KOMAExecuteOptions: Neue Anweisung 26
\KOMAoptions: Optionen aller KOMA-Familien abarbeiten 29
\KOMAProcessOptions: neu . . 17
\l@chapter: \sectfont durch Verwendung von Element chapterentry ersetzt . . . 188
Element chapterentrypagenumber wird verwendet 188
weitgehend umgeschrieben, um linksbündige Einträge zu ermöglichen 188
\l@part: \sectfont\large durch Verwendung von Element partentry ersetzt . 188
Element partentrypagenumber wird verwendet 188
\l@section: \sectfont durch Verwendung von Element sectionentry ersetzt . . . 189
Element sectionentrypagenumber wird verwendet 189
weitgehend umgeschrieben, um linksbündige Einträge zu ermöglichen 189
\lco@test: an neue Optionenabarbeitung angepasst . . . 290
\let@as@internal@defined: Herausreichen von internen Makros kann generell verhindert werden 18
\ltr@len@foldmarkthickness: Neu 251
\ltr@len@fromrulethickness: Neue Pseudolänge fromrulewidth 253
\maketitle: Vakatseiten in der Titelei immer im Seitenstil empty 154
internalonly: geht auch ohne Argument 17
toaddress: Neues Font-Element 250
locfield: vertauschte Bedeutung korrigiert 239
foldmark: Neu 249
toname: Neues Font-Element 250
footsepline: Option footinclude wird nicht mehr an typearea geschickt 135
specialmail: Neues Font-Element 250
\ps@headings: Unterscheidung auf twoside so spät wie möglich 139
\raggedchapterentry: neue Anweisung 189
\raggedsectionentry: neue Anweisung 190
\raggedsubject: neu (intern) 242

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

fromrule: Neu 249
\scr@fnt@wrn@pagehead: Warnung geändert 138
\scr@ifundefinedorrelax: vermurkste Funktion korrigiert 13
von \scr@ifundefined umbenannt 13
\scr@ifVTeX: umdefiniert von \scr@ifvtex 23
\scr@key@atlist: Neues Makro (intern) 28
\scr@key@name: Neues Makro (intern) 28
\scr@key@split@name@value: Neues Makro (intern) 28
\scr@key@value: Neues Makro (intern) 28
\scr@optiontest: neu (intern) 17
\scr@pti@ntest: neu (intern) 17
\scr@smashdp: neu (intern) . . 25
\scr@v@2.97c: Neues Macro 233
\specialmail@format: neu (intern) 265
\subject@format: neu (intern) 242
\subtitle: neue Möglichkeit 153
subtitle: neu (intern) 158
\useplength: \dimexpr statt \number verwendet, wenn ε-TEX verwendet wird 287

v2.97d

\@pt@@scan: \PackageInfo durch \PackageInfoNoLine ersetzt 59
a4paper: \PackageInfo durch \PackageInfoNoLine ersetzt 88
a5paper: \PackageInfo durch \PackageInfoNoLine ersetzt 88
b5paper: \PackageInfo durch \PackageInfoNoLine ersetzt 88
halfparskip: \PackageInfo durch \PackageInfoNoLine ersetzt 145
halfparskip*: \PackageInfo durch \PackageInfoNoLine ersetzt 145
halfparskip+: \PackageInfo durch \PackageInfoNoLine ersetzt 145
halfparskip-: \PackageInfo durch \PackageInfoNoLine ersetzt 145
landscape: \PackageInfo durch \PackageInfoNoLine ersetzt 88
parindent: \PackageInfo durch \PackageInfoNoLine ersetzt 145
parskip*: \PackageInfo durch \PackageInfoNoLine ersetzt 145
parskip+: \PackageInfo durch \PackageInfoNoLine ersetzt 145
parskip-: \PackageInfo durch \PackageInfoNoLine ersetzt 145
pdftex: \PackageInfo durch \PackageInfoNoLine ersetzt 88
headexclude: \PackageInfo durch \PackageInfoNoLine ersetzt 82
headnosepline: \PackageInfo durch \PackageInfoNoLine ersetzt 135
refvalue: Voreinstellung korrigiert 250
legalpaper: \PackageInfo durch \PackageInfoNoLine ersetzt 88
letterpaper: \PackageInfo durch \PackageInfoNoLine ersetzt 88
final: \PackageInfo durch \PackageInfoNoLine ersetzt 39
DIVcalc: \PackageInfo durch \PackageInfoNoLine ersetzt 83
DIVclassic: \PackageInfo durch \PackageInfoNoLine ersetzt 83
cleardoubleempty: \PackageInfo durch \PackageInfoNoLine

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

ersetzt 136
cleardoubleplain: \PackageInfo durch \PackageInfoNoLine ersetzt 136
cleardoublestandard: \PackageInfo durch \PackageInfoNoLine ersetzt 136
onecolumn: \PackageInfo durch \PackageInfoNoLine ersetzt 84
footexclude: \PackageInfo durch \PackageInfoNoLine ersetzt 82
footnosepline: \PackageInfo durch \PackageInfoNoLine ersetzt 135
portrait: \PackageInfo durch \PackageInfoNoLine ersetzt 88
\PackageInfoNoLine: neu . . . 25
mpexclude: \PackageInfo durch \PackageInfoNoLine ersetzt 82
\scr@v@2.97c: Wert geändert 233
\scr@v@2.97d: Neues Macro 233
Wert geändert 233
numericaldate: \PackageInfo durch \PackageInfoNoLine ersetzt 45
twoside: \PackageInfo durch \PackageInfoNoLine ersetzt 83
executivepaper: \PackageInfo durch \PackageInfoNoLine ersetzt 88
v2.97e
\@alignfrom: neue Werte für Verwendung der Absenderergänzung 235
\@firstheadfootfield: Optionen firsthead und firstfoot beachten 258
\@foldmarks: mfoldmarkvpos wird verwendet 255
\@hfoldmark: Neu (intern) . . 255
\@locfield: Der Absender kann nun ebenfalls in der Absenderergänzung stehen 266
\@usefolds: Neu (intern) . . . 240
\@vfoldmark: Neu (intern) . . 256
General: Die neuen Faltmarken werden berücksichtigt. . . . 294
firstfoot: Neu 243
firsthead: Neu 235
\isopaper: gewünschte Papierausrichtung wird beachtet 80
\ltr@len@bfoldmarklength: Neu 251
\ltr@len@foldmarkvpos: Neu 251
\ltr@len@hfoldmarklength: Neu 251
\ltr@len@lfoldmarkhpos: Neu 251
\ltr@len@lfoldmarklength: Neu 251
\ltr@len@locheight: Neu . . 252
\ltr@len@lochpos: Neu 252
\ltr@len@locvpos: Neu 252
\ltr@len@mfoldmarklength: Neu 251
\ltr@len@mfoldmarkvpos: Neu 251
\ltr@len@pfoldmarklength: Neu 251
\maybe@pair: funktioniert zum ersten Mal 91
onecolumn: auf vielfachen Wunsch reicht die Klasse die Option an das Paket weiter 84
oneside: auf vielfachen Wunsch reicht die Klasse die Option an das Paket weiter 83
\scr@key@split@name@value: verwendet \KV@@sp@def zur Entsorgung von Leerzeichen 28
\scr@removechar: Neu (intern) 241
\scr@v@2.97e: Neues Macro 233
\theequation: Kapitelnummer nur im Hauptteil 42
\thefigure: Kapitelnummer nur im Hauptteil 126
\thesection: Kapitelnummer nur im Hauptteil 169
\thetable: Kapitelnummer nur im Hauptteil 127
\vb@t@z: \long 255
twocolumn: auf vielfachen Wunsch reicht die Klasse die Option an das Paket weiter 84

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

twoside: auf vielfachen Wunsch reicht die Klasse die Option an das Paket weiter 83
\x@@paper: Optionen werden nach der neuen Syntax an typearea weitergeleitet 80
v2.98
General: first version based on lco files from Gernot Hassenplfug 313
\scr@@onefilewithoption: new helper macro 224
\scr@replacefile@msg: spellfix 225
\scr@v@2.98: Neues Macro . 233
v2.98a
\@locfield: bei zentriertem Absender wird die Absenderergänzung ggf. unter dem Logo ausgegeben 267
\scr@v@2.98a: Neues Macro 233
twocolumn: es wird die richtige Option an das Paket weitergereicht 84
v2.98b
\@locfield: Absenderergänzung immer setzen, wenn verwendet 266
\scr@v@2.98b: Neues Macro 233
v2.98c
\@bib@break: Neu (intern) . . 208
\@bigheadings: entfernt . . . 166
\@dblfpbot: in eigener scrsize-Datei definiert 63
\@dblfpsep: in eigener scrsize-Datei definiert 63
\@dblfptop: in eigener scrsize-Datei definiert 63
\@fpbot: in eigener scrsize-Datei definiert 63
\@fpsep: in eigener scrsize-Datei definiert 63
\@fptop: in eigener scrsize-Datei definiert 63
\@listi: \g@addto@macro durch \l@addto@macro ersetzt . 150
in eigener scrsize-Datei definiert 64
\@listii: \g@addto@macro durch \l@addto@macro ersetzt 150
in eigener scrsize-Datei definiert 64
\@listiii: \g@addto@macro durch \l@addto@macro ersetzt 150
in eigener scrsize-Datei definiert 64
\@listiv: in eigener scrsize-Datei definiert 64
\@listv: in eigener scrsize-Datei definiert 64
\@listvi: in eigener scrsize-Datei definiert 64
\@normalheadings: entfernt . 166
\@openbib@code: wird nun früher definiert 204
\@pt@@scan: Verwendung von \KOMA@UseObsoleteOption korrigiert 59
\@smallheadings: entfernt . . 166
General: Eigener scrsize-Datei definiert 60
Option headings=big zu den Voreinstellungen hinzugefügt 40
Umstellung für die pt-Dateien 150
tablecaptionabove: obsolete 111
tablecaptionbelow: obsolete 111
\AfterBibliographyPreamble: Neu 209
pagefoot: nicht länger ein Alias 138
pagehead: neuer Alias 138
pageheadfoot: neues Element 138
captions: Neuer Zentralschalter für caption 110
\AtEndBibliography: Neu . . 209
\bib@after@hook: Neu (intern) 208
\bib@before@hook: Neu (intern) 208
\bib@beginhook: Neu (intern) 204
\bib@endhook: Neu (intern) . 204
\bib@heading: wird nun früher definiert 205
\BreakBibliography: Neu 207, 208

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\c@pti@nb@sid@: vertikale Ausrichtung korrigiert 114
\ClassInfoNoLine: neu 25
\cleardoubleevenemptypage: neu 136
\cleardoubleevenplainpage: neu 136
\cleardoubleevenstandardpage: neu 136
\cleardoubleevenusingstyle: neu 136
\cleardoubleoddemptypage: neu 136
\cleardoubleoddplainpage: neu 136
\cleardoubleoddstandardpage: neu 136
\cleardoubleoddusingstyle: neu 136
\cleardoublepage: neue Voreinstellung für Versionen nach 2.98c 136
wird immer neu definiert . . 136
scr@dotchangeatdocument: Neu (intern) 161
\dblfloatsep: in eigener scrsize-Datei definiert . . . 63
idxtotoc: obsolete 211
headheight: neue Option 81
headings: Neue Option 163
headsepline: Option headinclude wird gesetzt, wobei kein Fehler ausgegeben wird, wenn sie nicht existiert. 135
\float@@listhead: neu (intern) 193
\float@listhead: \float@@listhead wird verwendet 200
\floatsep: in eigener scrsize-Datei definiert 63
\FN@mf@check: Neu (intern) . 133
\footfont: Neu 138
\footins: in eigener scrsize-Datei definiert 63
\footnotesep: in eigener scrsize-Datei definiert . . . 63
\footnotesize: \g@addto@macro durch \l@addto@macro ersetzt 150
in eigener scrsize-Datei definiert 61
\footref: Neue Anweisung . 134
chapteratlists: nicht mehr bei scrartcl 196
thebibliography: Neue Haken \bib@before@hook und \bib@after@hook eingefügt 207
\Huge: in eigener scrsize-Datei definiert 61
\huge: in eigener scrsize-Datei definiert 61
bibliography: Neu 205
bibtotoc: obsolete 206
bibtotocnumbered: obsolete . 206
\idx@@heading: neu (intern) 210
statt \idx@heading wird nun \idx@@heading umdefiniert 210
\idx@heading: Verwendung von \idx@@heading 211
\if@bib@break@: Neu (intern) 208
\if@tablecaptionabove: Definition erfolgt etwas früher 109
\if@tocleft: wird früher definiert 192
\ifonelinecaptions: Definition erfolgt etwas früher . . 109
bigheadings: obsolet 166
\intextsep: in eigener scrsize-Datei definiert 63
listof: Neue Option 193
listsindent: obsolet 196
listsleft: obsolet 196
liststotoc: obsolet 196
liststotocnumbered: obsolet 196
\KOMA@@@DeclareObsoleteOption: neu (intern) 37
\KOMA@@DeclareObsoleteOption: neu (intern) 37
\KOMA@iv@DeclareObsoleteOption: neu (intern) 37
\KOMAoption: Neue Anweisung 31
\LARGE: in eigener scrsize-Datei definiert 61
\Large: in eigener scrsize-Datei definiert 61
\large: in eigener scrsize-Datei definiert 61

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

cleardoublepage: verwendet \def statt \let, damit die Option open funktioniert . 136
smallheadings: obsolet 166
\multfootsep: Neu 134
\multiplefootnotemarker: Neu 134
\multiplefootnoteseparator: Neu 134
index: Neue Option 210
onelinecaption: keine Sonderbehandlung für caption2 mehr 111
obsolete 111
\newbibstyle: Neu 203
\newblock: wird nun früher definiert 204
\no@bib@break: Neu (intern) 208
\normalsize: in eigener scrsize-Datei definiert . . . 61
toc: Neue Option 192
tocindent: obsolet 193
tocleft: obsolet 193
pointednumbers: obsolet . . . 162
pointlessnumbers: obsolet . 162
fontsize: lädt nach Möglichkeit die zugehörige Schriftgrößendatei 60
noonelinecaption: keine Sonderbehandlung für caption2 mehr 111
obsolete 111
footnotes: Neue Option für Mehrfachfußnoten 129
footsepline: Option footinclude wird gesetzt, wobei kein Fehler ausgegeben wird, wenn sie nicht existiert. 135
normalheadings: obsolet . . . 166
\partopsep: in eigener scrsize-Datei definiert 63
open: Neue Option 160
openany: obsolet 160
openbib: obsolete 206
openright: obsolet 160
\sbltextfloatsep: in eigener scrsize-Datei definiert . . . 63
\scap@pos: Definition erfolgt etwas früher 109
\scap@valign: Definition erfolgt etwas früher 109
\scr@altsecnumhook: Neu (intern) 161
\scr@altsecnumhookfalse: Neu (intern) 161
\scr@altsecnumhooktrue: Neu (intern) 161
\scr@bibstyle@: Neu (intern) 204
\scr@bibstyle@oldstyle: Neu (intern) 204
\scr@bibstyle@openstyle: Neu (intern) 204
\scr@bibstylelist: Neu (intern) 204
\scr@caption@before@longtable@warning: neu (intern) 123
\scr@fnt@wrn@pagefoot: Neu (intern) 138
\scr@fnt@wrn@pagehead: Warnung geändert 138
\scr@footmisc@options: Neuer (intern) 130
\scr@footnotemark: Neu (intern) 130
\scr@mf@prepare: Neu (intern) 133
\scr@saved@footnotemark: Neu (intern) 130
\scr@saved@footnotetext: Neu (intern) 133
\scr@v@2.98c: Neues Macro 233
\scriptsize: in eigener scrsize-Datei definiert . . . 61
\small: \g@addto@macro durch \l@addto@macro ersetzt . 150
in eigener scrsize-Datei definiert 61
\textfloatsep: in eigener scrsize-Datei definiert . . . 63
\tiny: in eigener scrsize-Datei definiert 61
numbers: Neue Option 161
dvips: Ersatz geändert 88
Info statt Warnung 88
v2.9a
\@setcapmargin: \ifthispageodd statt \if@thispageodd . . 120
captionbeside: \ifthispageodd statt \if@thispageodd . . 114

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\setcapwidth: \ifthispageodd statt \if@thispageodd . . 118
v2.9d
letter: Fußnotenzähler wird korrekt zurückgesetzt . . . 276
v2.9h
\add@extra@listi: neu (intern) 150
\footnotesize: etwas robuster 150
\small: etwas robuster 150
v2.9i
\@usekomaname: optionales Argument 284
\@usekomavar: optionales Argument 284
\datenorsk: Workaround für den hyphen.cfg-Bug von Babel 55
\ifkomavarempty: Leerzeichen rausgeworfen 285
robust 285
\usekomavar: robust 284
v2.9j
\@usekomaname: \@firstofone als Defaultargument eingefügt 284
\@usekomavar: \@firstofone als Defaultargument eingefügt 284
v2.9k
General: es wird direkt in \@mainaux geschrieben . . 168
\l@chapter: Anpassung an Option tocleft 188
\l@part: Anpassung an Option tocleft 188
\l@section: Anpassung an Option tocleft 189
\scr@dottedtocline: auch das dritte Argument wird geändert 198
v2.9l
\@datefield: Berechnung von refhpos hinzugefügt 270
Berechnung von refwidth an die Dokumentation angepasst 270
\@firstheadfootfield: Positionierung verbessert 258
\deffootnote: \edef\@tempa ersetzt 132
\dictum: \edef durch \def ersetzt 41
\idx@@heading: \if@twoside durch \if@openright ersetzt 210
\idx@heading: \if@twoside durch \if@openright ersetzt 211
\ifthispagewasodd: neuer Schalter 43
\is@thispageodd: renamed \if@thispageodd 43
\ltr@len@refhpos: Neu 252
v2.9o
\@addrfield: \noindent eingefügt bei specialmail . . 260
\cc: \parfillskip angepasst 279
\encl: \parfillskip angepasst 279
thebibliography: Vergleich \bibpreamble mit \@empty korrigiert 207
\minisec: \nobreak nach dem Gruppenende behebt einen Bug im color Paket 182
v2.9p
\@@makechapterhead: \parfillskip auf \fill gesetzt 174
\@@makeschapterhead: \parfillskip auf \fill gesetzt 178
\@seccntformat: \protect eingefügt 170
captionbeside: \@temdima replaced by \wd\c@pti@nb@sid@b@x, because \@temdima would be changed 115
bibtotocnumbered: Nummerierung im Kolumnentitel . . 206
liststotocnumbered: Nummerierung im Kolumnentitel 196
v2.9q
\@@makeschapterhead: Klammerung an \@@makechapterhead angepasst 178
\@addmargin: \labelsep bleibt unverändert 216

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\cc: \parfillskip auf 1em plus 1fil geändert 279
\deffootnote: \ftn@font erlaubt aus Gründen der Kompatibilität mit dem *footnote* Paket kein Argument mehr 132
\dictum: zur Vermeidung von Problemen durch \raggeddictum Gruppe eingefügt 41
\encl: \parfillskip auf 1em plus 1fil geändert 279
v2.9r
\@processto: robust gegen leeren Empfänger 276
\cc: \parfillskip auf \z@ plus 1fil geändert 279
\closing: das Argument darf leer sein 278
Längenbegrenzung auf \hsize 278
\encl: \parfillskip auf \z@ plus 1fil geändert 279
\newcaptionname: Wenn die aktuelle Sprache, die geänderte ist, wird die Anderung sofort aktiviert. 45
\partmark: neu (für scrpage2) 174
\providecaptionname: Wenn die aktuelle Sprache, die geänderte ist, wird die Anderung sofort aktiviert. . . . 45
\renewcaptionname: Wenn die aktuelle Sprache, die geänderte ist, wird die Anderung sofort aktiviert. . . . 45
v2.9s
\LT@makecaption: vertikale Abstände durch verschieben von \vskip korrigiert . . . 123
v2.9t
\@maketitle: \par in Größe und Schrift des Titels 272
v2.9u
General: Der Fuß hat optional eine feste Position 16 mm über der unteren Papierkante. . 296
\captionsfinnish: Neu 48
Unterstützung für Finnisch (Dank Hannu Väisänen) . . 48
\datefinnish: Neu 48
version: Neue Option 232
\scr@compatibility: Neues Macro 232
\scr@v@2.9: Neues Macro . . 233
\scr@v@2.9t: Neues Macro . 233
\scr@v@first: Neues Macro 233
\scr@v@last: Neues Macro . 233
v3.00
\@@maybeautodot: neuerdings überflüssige \aftergroup entfernt 168
\@addpart: Notlösung für hyperref eingefügt 173
\@autodotfalse: arbeitet nun gobal 167
\@autodottrue: arbeitet nun gobal 167
\@chapter: jetzt auch Kapiteleinträge in andere float-Verzeichnisse mit hyperref 174
\@fontsizefilebase: Definition für scrextend vorgezogen 59
General: neues Paket 317
\addchap: jetzt auch Kapiteleinträge in andere float-Verzeichnisse mit hyperref 179
\areaset: Layout-Infos werden per \ta@layoutinfo in die log-Datei geschrieben . . . 106
\backmatter: \cleardoubleoddpage statt \cleardoublepage . 166
\cleardoubleevenpage: neu 136
\cleardoubleoddpage: neu . 136
\DefineFamily: Neue Anweisung 13
\DefineFamilyKey: Neue Anweisung 13
\DefineFamilyMember: Neue Anweisung 13
if@semitwoside: neuer Schalter (intern) 83
if@twoside: Schalter setzt auch \if@semitwoside 83
\Family@Numerical@Key: Neu (intern) 33
\Family@Option: Neu (intern) 28

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\FamilyBoolKey: Neue Anweisung 32
\FamilyElseValues: Neue Anweisung 31
\FamilyExecuteOptions: Neue Anweisung 25
\FamilyNumericalKey: Neue Anweisung 33
\FamilyOption: Neue Anweisung 28
\FamilyOptions: Neue Anweisung 26
\FamilyProcessOptions: Neue Anweisung 15
\FamilySetBool: Neue Anweisung 32
\float@listhead: Verwendung von \float@listhead ist nicht länger empfohlen . . 200
\frontmatter: \cleardoubleoddpage statt \cleardoublepage . 166
\ifscr@key@found@: umbenannt von \ifKOMA@key@found 28
\KOMAExecuteOptions: Kein optionales Argument mehr . . 26
\KOMAProcessOptions: kein optionales Argument mehr . . 17
\listoffigures: Verwendung von tocbasic 201
\listoflofname: neu für Paket tocbasic 201
\listoftables: Verwendung von tocbasic 201
\listoftocname: neu für Paket tocbasic 199
\listotlofname: neu für Paket tocbasic 201
\mainmatter: \cleardoubleoddpage statt \cleardoublepage . 166
\newcaptionname: nach scrbase verschoben 45
foldmarks: es muss \FamilyUnkownKeyVal statt \KOMA@unkown@keyval lokal umdefiniert werden . 240
\partpagestyle: nicht mehr in scrartcl definiert 142
\providecaptionname: nach scrbase verschoben 45
\renewcaptionname: nach scrbase verschoben 45
\scr@if@family: Neu (intern) 13
\scr@if@familymember: Neu (intern) 13
\scr@key@atlist: umbenannt von \KOMA@key@atlist . . . 28
\scr@key@name: umbenannt von \KOMA@key@name 28
\scr@key@split@name@value: umbenannt von \KOMA@key@split@name@value 28
\scr@key@value: umbenannt von \KOMA@key@value 28
\scr@redefine@newpage: new (internal) 229
\scr@saved@newpage: new (internal) 229
\scr@v@3.00: Neues Macro . 233
\ta@layoutinfo: neu (intern) 105
\tableofcontents: auf tocbasic umgestellt 200
\toc@heading: Verwendung von \toc@heading ist nicht länger empfohlen 199
\typearea: Layout-Infos werden per \ta@layoutinfo in die log-Datei geschrieben . . . 104
twoside: die Option kennt zusätzlich den Wert semi . 83

v3.01

\@listi: \@listi wird auf jeden Fall ausgeführt 150
\@subjfield: Verwendung der neuen Pseudolängen subjectbeforevskip und subjectaftervskip 272
\@topnewpage: Verwendung von \scr@float@addtolists@warning (aus scrkliof.dtx) 174
\addchap: Verwendung von \scr@float@addtolists@warning (aus scrkliof.dtx) 179
\ltr@len@subjectaftervskip: Neue Pseudolänge subjectaftervskip 274
\ltr@len@subjectbeforevskip: Neue Pseudolänge subjectbeforevskip 274

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\ltr@len@subjectvpos: Neue Pseudolänge subjectvpos 273
\scr@float@addtolists@warning: Neu (intern) 201
\scr@float@listhead@warning: Neu (intern) 201
\scr@v@3.01: Neues Macro . 233
v3.01a
\@footnotetext: long 133
a4paper: standard statt obsolet 88
a5paper: standard statt obsolet 88
b5paper: standard statt obsolet 88
tablecaptionabove: deprecated 111
tablecaptionbelow: deprecated 111
halfparskip: deprecated . . . 145
halfparskip*: deprecated . . 145
halfparskip+: deprecated . . 145
halfparskip-: deprecated . . 145
landscape: standard statt obsolet 88
parindent: deprecated 145
parskip*: deprecated 145
parskip+: deprecated 145
parskip-: deprecated 145
abstractoff: standard statt obsolet 153
abstracton: standard statt obsolet 153
scrdate: deprecated 45
pdftex: standard statt obsolet 88
idxtotoc: deprecated 211
headexclude: deprecated 82
headnosepline: deprecated . 135
legalpaper: standard statt obsolet 88
letterpaper: standard statt obsolet 88
bibtotoc: deprecated 206
bibtotocnumbered: deprecated 206
bigheadings: deprecated . . . 166
final: standard statt obsolet . 39
listsindent: deprecated . . . 196
listsleft: deprecated 196
liststotoc: deprecated 196
liststotocnumbered: deprecated 196
DIVcalc: standard statt obsolet 83
DIVclassic: standard statt obsolet 83
\KOMA@@@DeclareDeprecatedOption: neu (intern) 35
\KOMA@@@DeclareStandardOption: neu (intern) 36
\KOMA@@DeclareDeprecatedOption: neu (intern) 35
\KOMA@@DeclareStandardOption: neu (intern) 36
\KOMA@DeclareDeprecatedOption: neu (intern) 35
\KOMA@DeclareStandardOption: neu (intern) 36
\KOMA@UseDeprecatedOption: neu (intern) 36
\KOMA@UseStandardOption: neu (intern) 36
cleardoubleempty: deprecated 136
cleardoubleplain: deprecated 136
cleardoublestandard: deprecated 136
slocfield: deprecated 239
wlocfield: deprecated 239
smallheadings: deprecated . 166
onecolumn: standard statt obsolet 84
onelinecaption: deprecated 111
oneside: standard statt obsolet 83
noappendixprefix: deprecated 162
nochapterprefix: deprecated 162
tocindent: deprecated 193
tocleft: deprecated 193
pointednumbers: deprecated 162
pointlessnumbers: deprecated 162
noonelinecaption: deprecated 111
footexclude: deprecated 82
footnosepline: deprecated . 135
normalheadings: deprecated 166
portrait: standard statt obsolet 88
notitlepage: standard statt obsolet 152
openany: standard statt obsolet 160
openbib: standard statt obsolet 206
openright: standard statt obsolet 160
mpexclude: deprecated 82
orgdate: deprecated 45

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\scr@compatibility: Voreinstellung auf *last* geändert 232
\scr@saved@footnotetext: long 133
\scr@v@3.01a: Neues Macro 233
dvips: standard statt obsolet . 88
executivepaper: standard statt obsolet 88
v3.01b
General: Kompatibilitätseinstellungen werden in Paketen nur definiert, wenn sie noch nicht vorhanden sind 232
paper: Höhe und Breite vertauscht 89
\newcaptionname: missing make undefined 45
\providecaptionname: missing make undefined 45
\renewcaptionname: missing make undefined 45
\scr@ta@compatibility: Neues Macro 232
\scr@v@3.01b: Neues Macro 233
\typearea: Warnungen vor ungünstigen Einstellungen geändert 104
v3.01c
\datefinnish: Aktivierung korrigiert 48
\scr@v@3.01c: Neues Macro 233
v3.02
\@endpart: Verwendung des neuen \partheademptypage 171
labelinglabel: Neue Elemente labelinglabel und labelingseparator werden verwendet 215
neues Element 217
labelingseparator: neues Element 217
\captionsnorsk: Neu 48
\datenorsk: Neu 48
\parheademptypage: neue Konfigurationsmöglichkeit . . . 171
\scr@v@3.02: Neues Macro . 233
v3.02a
\@@makecaption: die Box selbst muss im \capfont gesetzt werden, falls darin eine Größenumschaltung stattfindet 123
diverse Gruppen- bzw. Argumentklammern ergänzt . . 121
\scr@smashdp: \smash nicht direkt verwenden, weil amsmath das inkompatibel umdefiniert 25
v3.02b
\scr@v@3.02b: Neues Macro 233
v3.02c
\@seascapefalse: neu 91
\@seascapetrue: neu 91
General: Fehlermeldung, wenn nicht ε-TEX verwendet wird 12
labelinglabel: labelinglabel und labelingseparator werden nun auch zur Berechnung der Breite verwendet 215
pagesize: neuer Wert false . 84
paper: neuer Wert seascape . 89
\if@seascape: neu 91
\ifnotundefined: ε-TEX ist zwingend 20
final: optionales Argument im Fall von scrextend gesetzt . 39
\KOMA@ifkey: definiert ebenfalls das Familienmitglied 33
\KOMA@ncmdkey: definiert ebenfalls das Familienmitglied . 34
\newcaptionname: undefined test improved 45
\providecaptionname: undefined test improved 45
\renewcaptionname: undefined test improved 45
\scr@ifundefinedorrelax: Sonderbehandlung für nicht ε-TEXentfernt 13
\scr@updatedvipspagesize: neu (intern) 85
\scr@v@3.02c: Neues Macro 233
\set@dvipspagesize: seascape wird beachtet 85
funktioniert auch im Dokument 85

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

v3.03
\@addrfield: Neue Anschriftmodi PP, backgroundimage, addresseeimage 260
Priorität kann Versandart ersetzen 260
\@afieldfalse: ersetzt durch \@afieldmode 237
\@afieldmode: neu (intern) . 237
\@afieldtrue: ersetzt durch \@afieldmode 237
General: Anschriftfeld enthält laut Doku die Rücksendeadresse, also machen wir das jetzt auch so 295
Bezeichner der Variablen fromzipcode wird gesetzt 294
erste Version des Pakets . . 323
set term of variablen fromzipcode 314
\AfterReadingMainAux: new 228
addresseeimage: Neu 280
addrfield: neue Werte 237
\if@afield: ersetzt durch \@afieldmode 237
\if@komavarempty: die Argumente werden nicht mehr direkt gelesen 285
die neue Anweisung \ifkomavar wird verwendet 285
\ifkomavar: Neu 285
zipcodeseparator: Neu 280
\ltr@len@PPdatamatrixvskip: Neu 252
\ltr@len@PPheadheight: Neu 252
\ltr@len@PPheadwidth: Neu 252
\ltr@priority: neu (intern) 238
PPcode: Neu 280
PPdata: Neues Font-Element 250
PPdatamatrix: Neu 280
PPlogo: Neues Font-Element 250
priority: neu 238
Neues Font-Element 250
prioritykey: Neues Font-Element 250
fromzipcode: Neu 280
\scr@hook@acma: new (internal) 230
\scr@saved@dofilelist: new 229
\scr@v@3.03: Neues Macro . 233
v3.03a
labelinglabel: Argumente in der Berechnung der Label-Breite korrigiert 215
\chaptermarkformat: Tilde durch \nobreakspace ersetzt, um die Kompatibilität mit Spanisch zu verbessern 169
\scr@v@3.03a: Neues Macro 233
v3.03b
\@addrfield: Schreibfehler bei »addressee« korrigiert . . 260
addresseeimage: Schreibfehler korrigiert 280
\ltr@len@toaddrheight: Implementierung an Doku angepasst 252
\scr@v@3.03b: Neues Macro 233
\thesection: \relax im Kompatibilitätstest durch ein Leerzeichen ersetzt, obwohl das die Wartbarkeit verschlechtert, aber hyperref hat mit dem \relax ein nicht dokumentieres Problem 169
v3.04
General: neue LCO NF von Jean-Marie Pacquet übernommen und ergänzt 293
neue LCO UScommercial9DW für Papier im Format letter oder legal in Umschlägen im Format commercial No. 9 mit zwei Fenstern 293
neue LCO UScommercial9 für Papier im Format letter oder legal in Umschlägen im Format commercial No. 9 mit einem Fenster auf Basis von http://www.envelopesuperstore.com/ 293
neue LCO USletter2w in Zusammenarbeit mit Engelbert Buxbaum und Richard Arès geschaffen 293
seit langem existierende LCO visualize zur Visualisierung

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

einiger Pseudolängen aufgenommen 293
\scr@v@3.04: Neues Macro . 233
v3.04a
abstract: die Unterscheidung, ob Titelseite oder Titelkopf sollte erst in der Umgebung und nicht schon bei deren Definition erfolgen 159
\maketitle: die Unterscheidung, ob Titelseite oder Titelkopf sollte erst in \maketitle und nicht schon bei dessen Definition erfolgen 154
\scr@v@3.04a: Neues Macro 233
v3.04b
General: Die Reihenfolge von Anweisungen und Optionen grundlegend geändert, um das Paket scrhack unabhängiger von der Reihenfolge beim Laden von Paketen zu machen. 323
\Family@Option: \edef durch \protected@edef ersetzt . 28
\FamilyExecuteOptions: \edef durch \protected@edef ersetzt 25
\scr@key@atlist und \scr@key@name vor Überschreiben in \serkeys geschützt 25
\ifstr: \edef durch \protected@edef ersetzt 20
hyperref: hyperref-Hack wird früher geladen 324
v3.05
\@firstheadfootfield: neue Pseudo-Länge firstfoothpos beachtet 258
neue Pseudo-Länge firstheadhpos beachtet 258
captionof: Neue Anweisung . 125
\l@part: Da auch article die Ebene -2 verwendet, wird das nun für scrartcl übernommen. 188
\ltr@len@firstfoothpos: Neu 253
\ltr@len@firstheadhpos: Neu 253
\pagemark: eine zusätzliche Gruppe eingefügt 138
\scr@v@3.05: Neues Macro . 233
\showfield@foot: neue Pseudo-Länge firstfoothpos beachtet 303
\showfield@head: neue Pseudo-Länge firstheadhpos beachtet 302
v3.05a
\addtokomafont: nach scrkbase verschoben 73
pagesize: neuer Wert dvipdfmx 84
\aliaskomafont: nach scrkbase verschoben 75
\newkomafont: nach scrkbase verschoben 75
\scr@v@3.05a: Neues Macro 233
\set@dvipdfmxpagesize: neu (intern) 85
\setkomafont: nach scrkbase verschoben 72
\usekomafont: nach scrkbase verschoben 74
\XdivY: Neu 18
\XmodY: Neu 18
v3.05b
\@addmargin: \partopsep muss auf 0 statt \parskip gesetzt werden, weil es zu \topsep addiert wird 216
v3.06
General: einschalten des tocbasic-Features chapteratlist in die Datei scrkliof.dtx verschoben 197
partentry: Verwendung von sectioning durch disposition ersetzt 183
\before@starttoc: statt \numberline wird \numberline@box umdefiniert 198
sectionentry: Verwendung von sectioning durch disposition ersetzt 183
\footnoterule: die neuen Längenmakros werden genutzt 130

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\ftn@rule@height: neu (intern) 131
\ftn@rule@test@values: neu (intern) 130
\ftn@rule@width: neu (intern) 131
chapterentry: Verwendung von sectioning durch disposition ersetzt 183
\if@at@twocolumn: einschalten des tocbasic-Features chapteratlist in die Datei scrkliof.dtx verschoben 176
listof: Alle über tocbasic realisierten Einstellungen wirken sich auch auf Verzeichnisse des Besitzers float aus, die erst später unter Kontrolle von tocbasic gestellt werden. 193
neuer Wert entryprefix . 194
\numberline: Verwendung von \numberline@prefix und \numberline@postfix . . 197
\numberline@box: neu (intern) 197
\numberline@numberformat: neu (intern) 197
\numberline@postfix: neu (intern) 197
\numberline@prefix: neu (intern) 197
\scr@numberline: nicht länger benötigt 198
\scr@v@3.06: Neues Macro . 233
\setfootnoterule: neue Anweisung 131

v3.06a
\@addrfield: Absatzparameter werden passend eingestellt 260
\@gen@firsthead: Verwendung des Font-Elements fromrule eingefügt 245
\cleardoubleevenusingstyle: \thispagestyle durch \pagestyle ersetzt 136
\cleardoubleoddusingstyle: \thispagestyle durch \pagestyle ersetzt 136
\cleardoubleusingstyle: \thispagestyle durch \pagestyle ersetzt 136

v3.07
\footnoterule: neues Font-Element footnoterule wird verwendet 130
footnoterule: footnoterule ist ein eneues Fontelement 131
\scr@v@3.07: Neues Macro . 233

v3.07a
\c@pti@nb@sid@: vertikale Ausrichtung im t- und b-Fall erneut korrigiert 114
\showfield@address: Implementierung der Höhe an die Änderung in scrlttr 3.03b angepasst. 303
\showfield@location: Implementierung der Höhe an die Änderung in scrlttr 3.03b angepasst. 303

v3.08
\@addpart: Verwendung von \addparttocentry 173
\@chapter: Verwendung von \addchaptertocentry . . 174
\@firstfoot: der Fuß des Briefbogens wird über eine Variable realisiert 253
\@firsthead: der Kopf des Briefbogens wird über eine Variable realisiert 245
\@gen@firsthead: der Kopf des Briefbogens wird über eine Variable realisiert 245
\@nextfoot: der Fuß der Folgeseiten wird über eine Variable realisiert 275
\@nexthead: der Kopf der Folgeseiten wird über eine Variable realisiert 274
\@part: Verwendung von \addparttocentry 171
\addchap: Verwendung von \addchaptertocentry . . 179
\addchaptertocentry: Neu . 178
\addparttocentry: Neu 173
\addtocentrydefault: Neu . 200
parskip: neuer Wert never . 145
\captionsnorsk: Neu (Dank Benjamin Hell) 48

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx

\datenorsk: Neu (Dank Benjamin Hell) 48
nextfoot: Neu 275
nexthead: Neu 274
\Family@@String@@Key: Neu (intern) 35
\Family@Option: leere Argumente in der Werteliste . . . 29
\Family@String@Key: Neu (intern) 35
\FamilyOptions: leere Argumente in der Optionenliste 27
\FamilyStringKey: Neue Anweisung 35
\firstfoot: der Fuß des Briefbogens wird über eine Variable realisiert 253
\firsthead: der Kopf des Briefbogens wird über eine Variable realisiert 245
firstfoot: Neu 253
firsthead: Neu 245
\nextfoot: der Fuß der Folgeseiten wird über eine Variable realisiert 275
\nexthead: der Kopf der Folgeseiten wird über eine Variable realisiert 274
\PreventPackageFromLoading: New 226
\ps@headings: \noindent added before usage of variable nexthead or nextfoot . . 140
\ResetPreventPackageFromLoading: New 227
\scr@firsthead@postsetvar: Neu (intern) 245
\scr@gobbleopt: New (internal) 228
\scr@v@3.08: Neues Macro . 233
\scrlfile@excludedpackages: New (internal) 226
\scrlfile@RequirePackage: New (internal) 228
\scrlfile@saved@RequirePackage: New (internal) 226
\setkomavar: \scr@*Name*@postsetname und \scr@*Name*@postsetvar 283
\StorePreventPackageFromLoading: New 227
v3.08a
\scr@gobbleopt: \@gobbleopt renamed due to compaibility problems with ntheorem . 228
\scr@v@3.08a: Neues Macro 233

File Key: a=scrkvers.dtx, b=scrkbase.dtx, c=scrkmisc.dtx, d=scrklang.dtx, e=scrkfont.dtx, f=scrktare.dtx, g=scrkfloa.dtx, h=scrkftn.dtx, i=scrkpage.dtx, j=scrkpar.dtx, k=scrktitl.dtx, l=scrksect.dtx, m=scrkliof.dtx, n=scrkbib.dtx, o=scrkidx.dtx, p=scrklist.dtx, q=scrlfile.dtx, r=scrlogo.dtx, s=scrkcomp.dtx, t=scrknpap.dtx, u=scrkvars.dtx, v=scrkplen.dtx, w=scrklco.dtx, x=japanlco.dtx, y=scrkcile.dtx, z=scrextend.dtx, A=scrhack.dtx