

The KOMA-pagestyle-package*

July 23, 2019

This is `scrpage.sty`. This obsolete package is part of the KOMA-script-bundle. It defines an user interface for pagestyle-definition. It's something like `fancyheadings` but not the same. You may use `scrpage` or `fancyhdr`. You may use `scrpage` with standard classes, too. Note: Package `scrpage` was replaced by package `scrpage2`. The manual for package `scrpage2` is part of the KOMA-Script guide.

Contents

1	Introduction	1
1.1	Interface for all users	1
1.2	Interface for experts	4
1.3	Configuration file	4
1.4	Example	4
2	Implementation	5
2.1	Options	5
2.2	Some Initialisation	6
2.3	Predefinitions	6
2.4	Expert-Pagestyle-Definition-Interface	6
2.5	User-Pagestyle-Definition-Interface	12
2.6	Width of head and foot	13
2.7	Fonts	15
2.8	Extended multilanguage formats	16
2.9	Configurationfile	16
2.10	End	16

*This file has version number v1.3e, last revised 2012/11/06.

1 Introduction

There is a simple user-interface and a expert-interface. Using the user-interface you can define a lot of different pagestyles. But there are combinations, you cannot define. Using the expert-interface you can do almost all you may ever want. To do more, you should learn more about defining macros yourself.

1.1 Interface for all users

<code>\headfont</code>	This font is used to write page heads and foots. You may change it using <code>\renewcommand</code> , e.g. <code>\renewcommand\headfont{\normalfont\slshape}</code> .
<code>\footfont</code>	This font is used to write page foots different from page heads. You may change it using <code>\renewcommand</code> , e.g. <code>\renewcommand\footfont{\normalfont\slshape}</code> . Only available at <code>scrpage2</code> .
<code>\headmark</code>	Using <code>twoside</code> -option this is <code>\leftmark</code> at left/even pages and <code>\rightmark</code> at right/odd pages. Using <code>oneside</code> -option there are only right pages, so it is <code>\rightmark</code> .
<code>\pnumfont</code>	This font is used to write pagenumbers at <code>\pagemark</code> . You may change it using <code>\renewcommand</code> (see <code>\headfont</code>).
<code>\pagemark</code>	This is the number of the actual page written with <code>\pnumfont</code> .
<code>\deftripstyle</code> <code>\deftripstyle*</code>	The command <code>\deftripstyle{<name>}[<olw>][<ilw>]{<headleft>}{<headmiddle>}{<headright>}{<footleft>}{<footmiddle>}{<footright>}</code> defines a new pagestyle. You can activate this pagestyle using <code>\pagestyle</code> or <code>\thispagestyle</code> . Without optional argument this defines a page like this:

<i>headright</i>	<i>headmiddle</i>	<i>headleft</i>	<i>headleft</i>	<i>headmiddle</i>	<i>headright</i>
left page			right page		
<i>footright</i>	<i>footmiddle</i>	<i>footleft</i>	<i>footleft</i>	<i>footmiddle</i>	<i>footright</i>

Using one optional argument [*ilw*] (means *inner line width*) you get a separationline between head and text and between text and foot like this (*ilw*=0.5pt):

<i>headright</i>	<i>headmiddle</i>	<i>headleft</i>	<i>headleft</i>	<i>headmiddle</i>	<i>headright</i>
left page			right page		
<i>footright</i>	<i>footmiddle</i>	<i>footleft</i>	<i>footleft</i>	<i>footmiddle</i>	<i>footright</i>

Using both optional arguments [*olw*] (means *outer line width*) and [*ilw*] you get a separationline between head and text and between text and foot and a line above head and below foot like this (*olw*=2pt, *ilw*=0.5pt):

The diagram illustrates the layout of a two-page spread. It consists of two vertical rectangles representing the pages. The left page is labeled "left page" and the right page is labeled "right page". At the top of each page, there is a horizontal bar representing the header. The left page header is divided into three sections: "headright", "headmiddle", and "headleft". The right page header is divided into three sections: "headleft", "headmiddle", and "headright". At the bottom of each page, there is a horizontal bar representing the footer. The left page footer is divided into three sections: "footright", "footmiddle", and "footleft". The right page footer is divided into three sections: "footleft", "footmiddle", and "footright".

You can define new pagestyles and you can redefine existing pagestyles. If you redefine an active pagestyle, this doesn't change the actual shown pagestyle. To activate you have to use always a `\pagestyle-` or `\thispagestyle-`command.

You can change the standard `pagestyle headings` to. If you want to change `pagestyle myheadings`, you should use the star-version of the command:

`\deftripstyle*{{name}}[{olw}][{ilw}]{{headleft}}{{headmiddle}}{{headright}}{{footleft}}{{footmiddle}}{{footright}}` this defines a my-version pagestyle. This means, section- and chapter-commands or perhaps `\tableofcontents` don't change the marks. Only `\markboth` and `\markright` change them. But if you don't use `\headmark`, `\leftmark` or `\rightmark`, star- and starless version of `\deftripstyle` are almost the same.

1.2 Interface for experts

`\defpagestyle` Command `\defpagestyle{⟨name⟩}{⟨head-definition⟩}{⟨foot-definition⟩}` defines a new pagestyle *name*. *head-definition* defines the head at the new pagestyle. It contains five parts:

(*toplinelength, toplinewidth*)
 {*evenpagehead*}{*oddpaghead*}{*onesidepagehead*}
 (*headseplinlength, headseplinewidth*)

These five parts have to be **one** argument! So you should put them together in group-braces (“{” and “}”).

	<i>foot-definition</i> analogous defines the foot at the new pagestyle.
<code>\newpagestyle</code>	Command <code>\newpagestyle{<name>}{<head-definition>}{<foot-definition>}</code> defines a real new pagestyle. If there's already a pagestyle named <i>name</i> , you'll get an error. Otherwise it's the same like <code>\defpagestyle</code> .
<code>\renewpagestyle</code>	Command <code>\renewpagestyle{<name>}{<head-definition>}{<foot-definition>}</code> redefines a old pagestyle. If there isn't a pagestyle named <i>name</i> , you'll get an error. Otherwise it's the same like <code>\defpagestyle</code> .
<code>\providepagestyle</code>	Command <code>\providepagestyle{<name>}{<head-definition>}{<foot-definition>}</code> defines a new pagestyle, if it is really new. If there is already a pagestyle named <i>name</i> , it does nothing (but writing an info to the log file). Otherwise it's the same like <code>\defpagestyle</code> . You may use the user-interface-macros <code>\pagemark</code> and <code>\headmark</code> , too.

1.3 Configuration file

There's a configuration file `scrpage.cfg`, included at the end of `scrpage.sty`, if it exists. There you can define your own default pagestyles using all the commands above.

1.4 Example

The pagestyles of this documentation was defined using:

```
\renewpagestyle{headings}{(\textwidth,1pt)%
    {\headmark\hfill}{\hfill\headmark}{\hfill\headmark\hfill}%
    (\textwidth,.4pt)}%
{(\textwidth,.4pt)%
    {\pagemark\hfill%
    Copyright \copyright\ Markus Kohm, 1994--2002}%
    {Package \texttt{scrpage}\hfill\pagemark}%
    {\rlap{Package \texttt{scrpage}}\hfill%
    Copyright \copyright\ Markus Kohm, 1994--2002\hfill%
    \llap\pagemark}%
    (\textwidth,1pt)}
\renewpagestyle{plain}{(\textwidth,1pt)%
    {\hfill}{\hfill}{\hfill}%
    (\textwidth,.4pt)}%
{(\textwidth,.4pt)%
    {\pagemark\hfill}%
    {\hfill\pagemark}%
    {\hfill\pagemark\hfill}%
    (\textwidth,1pt)}
\pagestyle{headings} % activate new version
```

2 Implementation

```
1 <*scrpage>
```

2.1 Options

`headinclude` `scrpage` knows some options. They are similar to the KOMA-script-classes.
`headexclude` To work with and without the KOMA-script-classes their definitions are not
`footinclude` quite simple.

```
footexclude 2 \DeclareOption{headinclude}{%
3   \PassOptionsToPackage{headinclude}{typearea}%
4 }
5 \DeclareOption{headexclude}{%
6   \PassOptionsToPackage{headexclude}{typearea}%
7 }
8 \DeclareOption{footinclude}{%
9   \PassOptionsToPackage{footinclude}{typearea}%
10 }
11 \DeclareOption{footexclude}{%
12   \PassOptionsToPackage{footexclude}{typearea}%
13 }
```

We do so, because we want so set `headinclude`, `footinclude` by default at old version (see below)!

Set the default options.

```
14 \ExecuteOptions{headinclude,footinclude}
```

But you may set other Options:

```
15 \ProcessOptions\relax
```

2.2 Some Initialisation

For easier handling of the differences between article-, report- and book-classes we define some more switches.

`\if@chapter` First distinguish between article and others.

```
16 \newif\if@chapter
17 \begingroup\expandafter\expandafter\expandafter\endgroup
18 \expandafter\ifx\csname chapter\endcsname\relax
19   \@chapterfalse
20 \else
21   \@chaptertrue
22 \fi
```

`\if@mainmatter` Next distinguish between matter- and nomatter-classes.

```
23 \def\@tempa{\newif\if@mainmatter\@mainmattertrue}
24 \begingroup\expandafter\expandafter\expandafter\endgroup
25 \expandafter\ifx\csname mainmatter\endcsname\relax
26 \else
27   \let\@tempa\relax
```

```

28 \fi
29 \@tempa

```

2.3 Predefinitions

There are some commands, you can use at pagestyle-definition.

```

\headmark This macro is \rightmark or \leftmark. But outside pagestyle-definition
           it's nothing.
           30 \let\headmark\relax

\pagemark This macro is the number of the page at the pagenumberfont:
           31 \DeclareRobustCommand\pagemark{\pnumfont\thepage}}

```

2.4 Expert-Pagestyle-Definition-Interface

The pagestyle-definition-interface for experts is not as easy as the simple interface we'll define later. But it's more flexible. We'll later use it to define the pagestyle-definition-interface for users.

```

\defpagestyle First we define the simple definition-interface. There's no test, if the
               pagestyle's defined twice.
               First there has to be the definition of head and foot.
               32 \def\defpagestyle{%
               We have to decide, whether it is a my-version or not:
               33 \ifstar
               34 {\@defpagestyle[-]}%
               35 {\@defpagestyle[+]}%

\@defpagestyle Now the we can define the head and the foot. Before version 2.5 we have
               distinguished one- and two-side definitions while loading the package and
               have had all the definitions of \@mkboth, \chaptermark, \sectionmark,
               and \subsectionmark at the definition of the page style. This was a lot of
               code:
               36 \if@twoside
               37 \def\@defpagestyle[#1]#2#3#4{%
               38 \expandafter\def\csname ps@#2\endcsname{%
               39 \def\@tempa{+}%
               40 \def\@tempb{#1}%
               41 \ifx\@tempa\@tempb
               42 \let\@mkboth\markboth
               43 \if@chapter
               44 \def\chaptermark####1{%
               45 \markboth {\ifnum \c@secnumdepth >\m@ne%
               46 \if@mainmatter\chaptermarkformat\fi\fi ####1}{%
               47 \ifnum \c@secnumdepth >\m@ne%
               48 \if@mainmatter\chaptermarkformat\fi\fi ####1}}%

```

```

49      \def\sectionmark####1{%
50          \markright {\ifnum \c@secnumdepth >\z@%
51              \sectionmarkformat\fi ####1}}%
52      \else
53          \def\sectionmark####1{%
54              \markboth {\ifnum \c@secnumdepth >\z@%
55                  \if@mainmatter\sectionmarkformat\fi\fi ####1}{%
56                  \ifnum \c@secnumdepth >\z@%
57                      \if@mainmatter\sectionmarkformat\fi\fi ####1}}%
58          \def\subsectionmark####1{%
59              \markright {\ifnum \c@secnumdepth >\@ne%
60                  \subsectionmarkformat\fi ####1}}%
61      \fi
62      \else
63          \let\@mkboth\@gobbletwo
64          \if@chapter
65              \def\chaptermark####1{%
66                  \def\subsectionmark####1{}%
67                  \fi
68              \def\sectionmark####1{}%
69          \fi
70          \def@twosidehead#3
71          \def@twosidefoot#4
72      }
73  }
74  }
75  \else
76      \def\@defpagestyle[#1]#2#3#4{%
77          \expandafter\def\csname ps@#2\endcsname{%
78              \def\@tempa{+}%
79              \def\@tempb{#1}%
80              \ifx\@tempa\@tempb
81                  \let\@mkboth\markboth
82                  \if@chapter
83                      \def\chaptermark####1{%
84                          \markright {\ifnum \c@secnumdepth >\m@ne%
85                              \if@mainmatter\chaptermarkformat\fi\fi ####1}}%
86                      \def\sectionmark####1{}%
87                  \else
88                      \def\sectionmark####1{%
89                          \markright{\ifnum \c@secnumdepth >\z@%
90                              \if@mainmatter\sectionmarkformat\fi\fi ####1}}%
91                      \def\subsectionmark####1{}%
92                  \fi
93              \else
94                  \let\@mkboth\@gobbletwo
95                  \if@chapter
96                      \def\chaptermark####1{}%
97                  \else

```

```

98         \def\subsectionmark####1{%
99         \fi
100        \def\sectionmark####1{%
101        \fi
102        \def@onesidehead#3
103        \def@onesidefoot#4
104    }
105 }
106 \fi

\def@twosidehead
\def@@twosidehead 107 \def\def@twosidehead(#1,#2)#3#4#5(#6,#7){%
108     \def\@evenhead{\let\headmark\leftmark%
109     \hss\hskip\@evenheadshift\vbox{\hsize=\@headwidth\relax%
110     \hf@rule{#1}{#2}{\@headwidth}%
111     \vskip#2
112     \vskip\baselineskip
113     \hb@xt@\@headwidth{{%
114         \headfont\strut #3}}}%
115     \hf@rule{#6}{#7}{\@headwidth}%
116 } \hskip\@oddheadshift\hss}%
117 \def\@oddhead{\let\headmark\rightmark
118     \hss\hskip\@oddheadshift\vbox{\hsize=\@headwidth\relax
119     \hf@rule{#1}{#2}{\@headwidth}%
120     \vskip#2%
121     \vskip\baselineskip
122     \hb@xt@\@headwidth{{%
123         \headfont\strut #4}}}%
124     \hf@rule{#6}{#7}{\@headwidth}%
125 } \hskip\@evenheadshift\hss}%
126 }
127 % \end{macro}
128 % \end{macro}
129 % \begin{macro}{\def@onesidehead}
130 %   \changes{v1.3}{2001/03/30}{Use of \cmd\@headwidth instead of
131 %     \cs{textwidth}}
132 % \begin{macro}{\def@@onesidehead}
133 % \begin{macro}{\def@@@onesidehead}
134 % Next the onesided head:
135 %   \begin{macrocode}
136 \def\def@onesidehead(#1,#2)#3#4#5(#6,#7){%
137     \def\@evenhead{}%
138     \def\@oddhead{\let\headmark\rightmark
139     \hss\hskip\@oddheadshift\vbox{\hsize=\@headwidth\relax
140     \hf@rule{#1}{#2}{\@headwidth}%
141     \vskip#2%
142     \vskip\baselineskip
143     \hb@xt@\@headwidth{{%
144         \headfont\strut #5}}}%

```

```

145     \hf@rule{#6}{#7}{\@headwidth}%
146   }\hskip\@evenheadshift\hss}%
147 }

```

\def@twosidefoot Next the twosided foot:

```

\def@twosidefoot 148 \def\def@twosidefoot(#1,#2)#3#4#5(#6,#7){%
\def@@twosidefoot 149   \def\@evenfoot{\let\headmark\leftmark
150     \hss\hskip\@evenfootshift\ vbox{\hsize=\@footwidth\relax
151       \topfoot@rule{#1}{#2}\hb@xt@\@footwidth{\%
152         \headfont\footfont\strut #3}}%
153       \botfoot@rule{#6}{#7}}\hskip\@oddfootshift\hss}%
154   \def\@oddfoot{\let\headmark\rightmark
155     \hss\hskip\@oddfootshift\ vbox{\hsize=\@footwidth\relax
156       \topfoot@rule{#1}{#2}\hb@xt@\@footwidth{\%
157         \headfont\footfont\strut #4}}%
158       \botfoot@rule{#6}{#7}}\hskip\@evenfootshift\hss}%
159 }

```

\def@onesidefoot

```

\def@onesidefoot 160 \def\def@onesidefoot(#1,#2)#3#4#5(#6,#7){%
\def@@onesidefoot 161   \def\@evenfoot{}%
162   \def\@oddfoot{\let\headmark\rightmark
163     \hss\hskip\@oddfootshift\ vbox{\hsize=\@footwidth\relax
164       \topfoot@rule{#1}{#2}\hb@xt@\@footwidth{\%
165         \headfont\footfont\strut #5}}
166       \botfoot@rule{#6}{#7}}\hskip\@evenfootshift\hss}%
167 }

```

We now define the rules used at head and foot.

\topfoot@rule

```

168 \newcommand\topfoot@rule[2]{%
169   \@tempdima\baselineskip\advance\@tempdima by-.7\normalbaselineskip
170   \advance\@tempdima by -#2
171   \vskip\@tempdima\hf@rule{#1}{#2}{\@footwidth}%
172   \vskip-\@tempdima}

```

\botfoot@rule

```

173 \newcommand\botfoot@rule[2]{%
174   \@tempdima-\baselineskip\advance\@tempdima by .3\normalbaselineskip
175   \advance\@tempdima by #2
176   \vskip\@tempdima\hf@rule{#1}{#2}{\@footwidth}%
177 }

```

\hf@rule

```

178 \newcommand\hf@rule[3]{%
179   \setlength{\@tempdimb}{#1}%
180   \setlength{\@tempdimb}{.5\@tempdimb}%
181   \hb@xt@#3{%

```

```

182     \hfill%
183     \llap{\vrule\@depth#2\@height\z@\@width\@tempdimb}%
184     \rlap{\vrule\@depth#2\@height\z@\@width\@tempdimb}%
185     \hfill%
186 }%
187 }

```

Using these definitions, we can define the testing macros.

`\newpagestyle` First the simple star/starless-selection:

```
188 \def\newpagestyle{%
```

We have to decide, whether it is a my-version or not:

```

189 \ifstar
190 {\@newpagestyle[-]}%
191 {\@newpagestyle[+]}%

```

`\@newpagestyle` There we have to distinguish, whether the pagestyle is already defined or not.

```

192 \def\@newpagestyle[#1]#2#3#4{%
193   \expandafter\ifx\csname ps@#2\endcsname\relax
    Then we can use the already defined \@defpagestyle.
194     \@defpagestyle[#1]{#2}{#3}{#4}%
195   \else
196     \PackageError
197       {scrpage}%
198       {Your command was ignored}%
199       {There is already a pagestyle named ‘‘#1’’. \MessageBreak%
200        Use \protect\defpagestyle\space, \protect\renewpagestyle or
201        \protect\providepagestyle.}%
202   \fi
203 }

```

`\renewpagestyle` First the simple star/starless-selection:

```
204 \def\renewpagestyle{%
```

We have to decide, whether it is a my-version or not:

```

205 \ifstar
206 {\@renewpagestyle[-]}%
207 {\@renewpagestyle[+]}%

```

`\@renewpagestyle` There we have to distinguish, whether the pagestyle is already defined or not.

```

208 \def\@renewpagestyle[#1]#2#3#4{%
209   \begingroup\expandafter\expandafter\expandafter\endgroup
210   \expandafter\ifx\csname ps@#2\endcsname\relax
211     \PackageError
212       {scrpage}%
213       {Your command was ignored}%

```

```

214      {There is no pagestyle named ‘‘#1’’. \MessageBreak%
215      Use \protect\defpagestyle, \protect\newpagestyle or
216      \protect\providepagestyle.}%
217  \else
    Then we can use the already defined \@defpagestyle.
218      \@defpagestyle[#1]{#2}{#3}{#4}%
219  \fi
220 }
```

`\providepagestyle` First the simple star/starless-selection:

```

221 \def\providepagestyle{%
    We have to decide, whether it is a my-version or not:
222  \@ifstar
223  {\@providepagestyle[-]}%
224  {\@providepagestyle[+]}}
```

`\@providepagestyle` There we have to distinguish, whether the pagestyle is already defined or not.

```

225 \def\@providepagestyle[#1]#2#3#4{%
226  \expandafter\ifx\csname ps@#2\endcsname\relax
    Then we can use the already defined \@defpagestyle.
227      \@defpagestyle[#1]{#2}{#3}{#4}%
228  \else
229      \PackageInfo
230      {scrpage}%
231      {\protect\providepagestyle\protect{#1\protect} ignored.}%
232  \fi
233 }
```

2.5 User-Pagestyle-Definition-Interface

The pagestyle-definition-interface for users is much easier than the expert-version. Using the expert-macros, the definition is easy, too.

`\deftripstyle` At version 1 we define the selection of star- or starless-version first:

```

234 \def\deftripstyle{%
235  \@ifstar
236  {\@deftripstyle[-]}%
237  {\@deftripstyle[+]}}
```

Next we check, if there is an optional argument:

```

\@deftripstyle
238 \def\@deftripstyle[#1]#2{%
239  \@ifnextchar[%]
240  {\@@deftripstyle[#1]{#2}}%
241  {\@@deftripstyle[#1]{#2}[\z@]}}
```

or perhaps two optional arguments:

`\@@deftripstyle`

```
242 \def\@@deftripstyle[#1]#2[#3]{%
```

Ok, now we have the problem, that there can be only 9 arguments. So we have to distinguish the star-versions by a macro:

```
243 \def\@tempa{#1}%
244 \@ifnextchar[%
245 {\@@@deftripstyle#2[#3]}%
246 {\@@@deftripstyle#2[\z@] [#3]}}
```

Now the main-definition:

`\@@@deftripstyle`

```
247 \def\@@@deftripstyle#1[#2] [#3]#4#5#6#7#8#9{%
248 \def\@tempb{+}%
249 \ifx\@tempa\@tempb
250 \defpagestyle{#1}%
251 {(\@headwidth,#2)%
252 {\rlap{#6}\hfill{#5}\hfill\llap{#4}}%
253 {\rlap{#4}\hfill{#5}\hfill\llap{#6}}%
254 {\rlap{#4}\hfill{#5}\hfill\llap{#6}}%
255 (\@headwidth,#3)}%
256 {(\@footwidth,#3)%
257 {\rlap{#9}\hfill{#8}\hfill\llap{#7}}%
258 {\rlap{#7}\hfill{#8}\hfill\llap{#9}}%
259 {\rlap{#7}\hfill{#8}\hfill\llap{#9}}%
260 (\@footwidth,#2)}%
261 \else
262 \defpagestyle*{#1}%
263 {(\@headwidth,#2)%
264 {\rlap{#6}\hfill{#5}\hfill\llap{#4}}%
265 {\rlap{#4}\hfill{#5}\hfill\llap{#6}}%
266 {\rlap{#4}\hfill{#5}\hfill\llap{#6}}%
267 (\@headwidth,#3)}%
268 {(\@footwidth,#3)%
269 {\rlap{#9}\hfill{#8}\hfill\llap{#7}}%
270 {\rlap{#7}\hfill{#8}\hfill\llap{#9}}%
271 {\rlap{#7}\hfill{#8}\hfill\llap{#9}}%
272 (\@footwidth,#2)}%
273 \fi
274 }
```

2.6 Width of head and foot

<code>\setheadwidth</code>	Until version 1.3, head and foot of a page had same width like the text
<code>\setfootwidth</code>	block. Since version 1.3 this was changed. Now the head and foot have
<code>\set@hf@width</code>	their own width and an optional shift relative to the text part of the page.
<code>\settowidthof</code>	This shift can be positive or negative. We set the width and shift with
<code>\deftowidthof</code>	

one macro and save them at macros. The set macros have an optional argument: the shift.

For definition of symbolic length values there is the macro `\settowidthof`. And to define a macro with this value, there is `\deftowidthof`.

```

275 \newcommand*{\settowidthof}[2]{%
276   \edef\@tempa{#2}%
277   \edef\@tempb{paper}\ifx\@tempa\@tempb
278     \setlength{#1}{\paperwidth}%
279   \else\edef\@tempb{text}\ifx\@tempa\@tempb
280     \setlength{#1}{\textwidth}%
281   \else\edef\@tempb{textwithmarginpar}\ifx\@tempa\@tempb
282     \setlength{#1}{\textwidth}%
283     \addtolength{#1}{\marginparwidth}%
284     \addtolength{#1}{\marginparsep}%
285   \else\edef\@tempb{head}\ifx\@tempa\@tempb
286     \setlength{#1}{\@headwidth}%
287   \else\edef\@tempb{foot}\ifx\@tempa\@tempb
288     \setlength{#1}{\@footwidth}%
289   \else\edef\@tempb{headtopline}\ifx\@tempa\@tempb
290     \setlength{#1}{\scr@headabove@linelength}%
291   \else\edef\@tempb{headsepline}\ifx\@tempa\@tempb
292     \setlength{#1}{\scr@headbelow@linelength}%
293   \else\edef\@tempb{footsepline}\ifx\@tempa\@tempb
294     \setlength{#1}{\scr@footabove@linelength}%
295   \else\edef\@tempb{footbotline}\ifx\@tempa\@tempb
296     \setlength{#1}{\scr@footbelow@linelength}%
297   \else\edef\@tempb{page}\ifx\@tempa\@tempb
298     \setlength{#1}{\paperwidth}%
299     \begingroup\expandafter\expandafter\expandafter\endgroup
300     \expandafter\ifx\csize ta@bcor\endcsize\relax\else
301       \addtolength{#1}{-\ta@bcor}%
302     \fi
303   \else
304     \setlength{#1}{#2}%
305   \fi
306 \fi
307 \fi
308 \fi
309 \fi
310 \fi
311 \fi
312 \fi
313 \fi
314 \fi
315 }
316 \newcommand*{\deftowidthof}[2]{%
317   \edef\@tempa{#2}\edef\@tempb{autohead}\ifx\@tempa\@tempb
318     \expandafter\def\csize #1\endcsize{\@headwidth}%

```

```

319 \else\edef\@tempb{autofoot}\ifx\@tempa\@tempb
320 \expandafter\def\csname #1\endcsname{\@footwidth}%
321 \else
322 \settowidthof{\@tempdima}{#2}%
323 \expandafter\edef\csname #1\endcsname{\the\@tempdima}%
324 \fi
325 \fi
326 }
327 \newcommand*\setheadwidth}[2][\@empty]{%
328 \set@hf@width{head}{#1}{#2} %
329 }
330 \newcommand*\setfootwidth}[2][\@empty]{%
331 \set@hf@width{foot}{#1}{#2}%
332 }
333 \newcommand*\set@hf@width}[3]{%
334 \settowidthof{\@tempdima}{#3}%
335 \ifdim\@tempdima>\paperwidth
336 \PackageWarning
337 {scrpage}%
338 {You've set width of #1 to a value\MessageBreak
339 greater than width of page!\MessageBreak%
340 I'll reduce width of #1 to width of page}%
341 \setlength{\@tempdima}{\paperwidth}%
342 \else
343 \ifdim\@tempdima<\z@
344 \PackageWarning
345 {scrpage}%
346 {You've set width of #1 to negative value!\MessageBreak%
347 I'll set it to 0pt}%
348 \setlength{\@tempdima}{\z@}%
349 \fi
350 \fi
351 \expandafter\edef\csname @#1width\endcsname{\the\@tempdima}%
352 \edef\@tempa{#2}%
353 \setlength{\@tempdima}{\textwidth}%
354 \addtolength{\@tempdima}{-\csname @#1width\endcsname}%
355 \ifx\@tempa\@empty
356 \if@twoside
357 \setlength{\@tempdimb}{.333333333\@tempdima}%
358 \else
359 \setlength{\@tempdimb}{.5\@tempdima}%
360 \fi
361 \else
362 \setlength{\@tempdimb}{#2}%
363 \fi
364 \addtolength{\@tempdima}{-\@tempdimb}%
365 \ifdim\@tempdimb<\z@
366 \setlength{\@tempdimb}{-\@tempdimb}%
367 \expandafter\edef\csname @odd#1shift\endcsname{-\the\@tempdimb}%

```

```

368 \else
369   \expandafter\edef\csname @odd#1shift\endcsname{\the\@tempdimb}%
370 \fi
371 \ifdim\@tempdima<\z@
372   \setlength{\@tempdima}{-\@tempdima}%
373   \expandafter\edef\csname @even#1shift\endcsname{-\the\@tempdima}%
374 \else
375   \expandafter\edef\csname @even#1shift\endcsname{\the\@tempdima}%
376 \fi
377 }

```

```

\@headwidth The values of width and shift of head and foot are saved at internal macros.
\@footwidth 378 \newcommand*{\@headwidth}{\textwidth}%
\@oddheadshift 379 \newcommand*{\@oddheadshift}{\z@}%
\@evenheadshift 380 \newcommand*{\@evenheadshift}{\z@}%
\@oddfootshift 381 \newcommand*{\@footwidth}{\textwidth}%
\@evenfootshift 382 \newcommand*{\@oddfootshift}{\z@}%
383 \newcommand*{\@evenfootshift}{\z@}%

```

2.7 Fonts

`\pnumfont` We have to define a macro defining the font, used by `\pagemark`. Because of this macro may already be defined by a KOMA-script-class, we have to do this, using `\providecommand`

```
384 \providecommand*{\pnumfont}{\normalfont}
```

`\headfont` There is also a macro used by `\@evenhead`, `\@oddhead` and `\@evenfoot`, `\@oddfoot`:

```

385 \providecommand*{\headfont}{\normalfont}%
386 }

```

2.8 Extended multilanguage formats

```

\partmarkformat There are some macros to handle languages like Chinese, Japanese or Ko-
\chaptermarkformat rean. These macros were defined first by Werner Lemberg at his CJK-
\sectionmarkformat bundle. See scrclass.dtx for further information.
\subsectionmarkformat 387 \providecommand{\partmarkformat}          {\partname\ \thepart. \ }
388 \if@chapter
389   \providecommand{\chaptermarkformat}      {\@chapapp\ \thechapter. \ }
390   \providecommand{\sectionmarkformat}      {\thesection. \ }
391 \else
392   \providecommand{\sectionmarkformat}      {\thesection\quad}
393   \providecommand{\subsectionmarkformat}   {\thesubsection\quad}
394 \fi

```

2.9 Configurationfile

Not last and not least we include the local configuration-file `scrpage.cfg`, if it exists.

```
395 \InputIfFileExists{scrpage.cfg}
396     {\typeout{*****^J%
397             * Local config file scrpage.cfg used^J%
398             *****}}
399     {}
```

2.10 End

`\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.

```
400 \@ifundefined{KOMAScript}{%
401   \DeclareRobustCommand{\KOMAScript}{\textsf{K\kern.05em O\kern.05em%
402     M\kern.05em A\kern.1em-\kern.1em Script}}}{%
403 \scrpage}
```