

The `sidecap` package

Rolf Niepraschk (niepraschk@ptb.de) Hubert Gäßlein

v1.7a – 2023/01/24

Abstract

This package defines the new environments `SCfigure` and `SCTable`, analogous to `figure` and `table`, which make it easy to typeset captions sideways.

Additionally, a `wide` environment is defined; it allows to use the margin area, e. g., for figures wider than `\textwidth`.

1 Introduction

In some cases it may be useful to typeset the caption aside the figure or the table. For this purpose the package `sidecap` defines the new environments `SCfigure` and `SCTable`. The figure or the table and the caption are put into two minipages that are positioned side by side and centered as a whole. The space between the minipages is `\sidecaptionsep`. The correct positioning with respect to left and right pages requires at least two compilation runs.

2 Usage

```
\usepackage[<option>]{sidecap}
\begin{SCTable} [<relwidth>][<float>] ... \end{SCTable}
\begin{SCfigure} [<relwidth>][<float>] ... \end{SCfigure}
\begin{SCTable*} [<relwidth>][<float>] ... \end{SCTable*}
\begin{SCfigure*} [<relwidth>][<float>] ... \end{SCfigure*}
```

<option> – `outercaption` (default): Caption appears left on left pages and right on right pages.

`innercaption`: Caption appears right on left pages and left on right pages.

`leftcaption`, `rightcaption`: Caption is always on the left or right, respectively.

`wide`: The floating objects may extend into the margin area.

`raggedright`, `raggedleft`, `ragged`: Better justification for small captions. The `ragged2e` package is used if available.

<relwidth> – optional; caption width relative to the width of the figure or table. A large value (e.g., 50) reserves the maximum width that is possible. Default is `\sidecaptionrelwidth` (which is initialized to 1.0).

<float> – optional; like the floating position parameter of the original table/figure environments. Default is `tbp`.

```
\begin{wide} ... \end{wide}
```

The `wide` environment may be used inside `figure` and `table` environments as well as in the normal text.

3 Required packages

This package requires the standard L^AT_EX package `ifthen`.

4 Supported packages

This package is compatible with L^AT_EX package `hyperref` (tested with version 6.71v as of 2001/11/12). If the `raggedright` or `raggedleft` or `ragged` package option has been given, then captions will be set with ragged margin. The `ragged2e` package will be used if it can be found.

5 The implementation

5.1 Register allocation and auxiliary macros

```

1 (*package)
2
3 \RequirePackage{ifthen}
4
5 \@ifdefinable{\SC@BOXWD}{\newdimen\SC@BOXWD}
6 \@ifdefinable{\SC@CAPWD}{\newdimen\SC@CAPWD}
7 % 2 noetig ???
8 \@ifdefinable{\SC@tempdima}{\newdimen\SC@tempdima}
9 \@ifdefinable{\SC@tempdimb}{\newdimen\SC@tempdimb}
10 \newcounter{SC@C} \newsavebox{\SC@BOX} \newcommand*{\SC@FLOAT}{}
11 \newcommand*{\SC@IDENT}{\SC@number\value{SC@C}}
12 \newcommand*\SC@label[1]{\gdef\SC@LABtext{#1}\ignorespaces}
13 \newcommand*\SC@orig@caption{} \newcommand*\SC@orig@label{}
14 \newcommand*\sidecaptionsep{}
15 \edef\sidecaptionsep{\the\marginparsep}
16 \newcommand*\sidecaptionrelwidth{1.0}
17 \newcommand*\sidecaptionvpos[2]{% 2002/08/31
18   \@namedef{SC@#1@vpos}{#2}%
19 }
20
21 \newboolean{SC@wide}
22 \def\SC@widefalse{\global\let\ifSC@wide\iffalse}
23 \def\SC@widetrue {\global\let\ifSC@wide\iftrue}
24
25 % Kann man auf dieses Laengenregister verzichten? Nein! (RN) <****>
26
27 \newcommand*\SC@calcwidths[1]{%
28   \SC@tempdima#1\relax
29   \SC@BOXWD\wd\SC@BOX
30   \SC@CAPWD\SC@fraction\SC@BOXWD
31   \@tempdima\SC@BOXWD
32   \advance\@tempdima\SC@CAPWD

```

```

33 \advance\@tempdima\sidecaptionsep
34 \ifdim\@tempdima>\SC@tempdima
35   \advance\SC@CAPWD-\@tempdima
36   \advance\SC@CAPWD\SC@tempdima
37 \fi
38 \SC@tempdimb\@tempdima
39 }
40
41 %\newcommand*\isSC@WIDEi}[2]{#2} %default: false
42 %\newcommand*\isSC@WIDEii}[2]{#2} %default: false
43 \newcommand*\isSC@WIDEi}[2]{\ifSC@wide#1\else#2\fi} %default: false
44 \newcommand*\isSC@WIDEii}[2]{\ifSC@wide#1\else#2\fi} %default: false
45
46 %\newcommand*\isSC@First}[2]{#1} %default: true
47
48 \newcommand*\SC@justify{}
49 \newcommand*\SC@justify@caption{}
50 \newcommand*\SC@justify@body{}
51
52 \newboolean{SC@origRagged} \setboolean{SC@origRagged}{false} %% ???

\SC@SaveCommands
53 \newcommand*\SC@SaveCommands){%
54   \let\SC@orig@caption=\caption \let\SC@orig@label=\label%
55 }

56 \SC@SaveCommands % necessary? or even wrong?

```

5.2 Package hyperref compatibility

Note: The `\pageref` doesn't work with `hyperref` ...:-(

```

57 \if@twoside
58   \newcommand*\isSC@ODD}[2]{%
59     \ifthenelse{\isodd{\pageref{\SC@IDENT}}}{#1}{#2}}
60 \else
61   \newcommand*\isSC@ODD}[2]{#1}%
62 \fi

```

5.3 Option processing

```

63 \DeclareOption{innercaption}{%
64   \renewcommand*\SC@FLOAT}[2]{%
65     \isSC@ODD{#1}\hspace{\sidecaptionsep}{#2} {{#2}\hspace{\sidecaptionsep}{#1}}}
66   \renewcommand*\SC@justify@caption{%
67     \isSC@ODD{\RaggedLeft}{\RaggedRight}}
68 }
69
70 \DeclareOption{outercaption}{%
71   \renewcommand*\SC@FLOAT}[2]{%
72     \isSC@ODD{#2}\hspace{\sidecaptionsep}{#1} {{#1}\hspace{\sidecaptionsep}{#2}}}
73   \renewcommand*\SC@justify@caption{%
74     \isSC@ODD{\RaggedRight}{\RaggedLeft}}
75 }
76

```

```

77 \DeclareOption{rightcaption}{%
78   \renewcommand*{\SC@FLOAT}[2]{\#2\hspace{\sidecaptionsep}{\#1}}
79   \def\SC@justify@caption{\RaggedRight}%
80 }
81
82 \DeclareOption{leftcaption}{%
83   \renewcommand*{\SC@FLOAT}[2]{\#1\hspace{\sidecaptionsep}{\#2}}
84   \def\SC@justify@caption{\RaggedLeft}%
85 }
86 \DeclareOption{wide}{%
87   \renewcommand*{\isSC@WIDEi}[2]{\if@twocolumn #2\else #1\fi}% ???
88   \renewcommand*{\isSC@WIDEii}[2]{\#1}}

```

These ‘ragged...’ options should stay here, behind the ‘...caption’ options!

```

89 \DeclareOption{raggedright}{%
90   \def\SC@justify{\RaggedRight}}
91 \DeclareOption{raggedleft}{%
92   \def\SC@justify{\RaggedLeft}}
93 \DeclareOption{ragged}{%
94   \def\SC@justify{\SC@justify@caption}%   \let ?
95 }
96 \DeclareOption{origragged}{\setboolean{SC@origRagged}{true}}
97 \DeclareOption{margincaption}{%
98   \ds@outercaption\ds@outerbody\ds@wide
99   \let\ds@innercaption\@empty
100  \let\ds@leftcaption\@empty
101  \let\ds@rightcaption\@empty
102  \def\SC@calcwidths#1{%
103    \SC@tempdimb#1\relax
104    \SC@BOXWD\SC@tempdimb % *-form !!! ???
105    \advance\SC@BOXWD-\marginparwidth
106    \advance\SC@BOXWD-\marginparsep
107    \SC@CAPWD\marginparwidth
108  }%
109 }
110 \DeclareOption{innerbody}{%
111   \def\SC@justify@body{%
112     \isSC@ODD{\RaggedRight}{\RaggedLeft}}
113 }
114
115 \DeclareOption{outerbody}{%
116   \def\SC@justify@body{%
117     \isSC@ODD{\RaggedLeft}{\RaggedRight}}
118 }
119
120 \DeclareOption{centerbody}{%
121   \def\SC@justify@body{\Centering}
122 }
123
124 \DeclareOption{rightbody}{%
125   \def\SC@justify@body{\RaggedLeft}%
126 }

```

```

127
128 \DeclareOption{leftbody}{%
129   \def\SC@justify@body{\RaggedRight}%
130 }

131 \ExecuteOptions{outercaption}
132 %\InputIfFileExists{sidecap.cfg}{-}{-} % hier ? % 2002/06/14
133 \ProcessOptions\relax

134 \AtBeginDocument{%
135   \ifSC@origRagged\else
136     \ifpackageloaded{ragged2e}{-}%
137     {%
138       \IfFileExists{ragged2e.sty}{%
139         \RequirePackage[NewParameters]{ragged2e}%
140         {%
141           \PackageWarningNoLine{sidecap}{%
142             Package 'ragged2e' not found -- expect bad formatting}%
143           }%
144         }
145       \fi
146     \@ifundefined{RaggedRight}%
147     {%
148       \let\RaggedLeft\raggedleft
149       \let\RaggedRight\raggedright
150       \let\Centering\centering
151     }{}
152 }

```

5.4 User-level macros (environments)

SCfigure (*env.*) Simply passes the first (optional) parameter and the required one, in this case ‘figure’, to **SC@float**. The figure caption should be bottom aligned.

```

153
154 %\newcommand*{SCfigureVpos{b}}% 2002/06/14
155 \@ifdefinable{SC@figure@vpos}{\def\SC@figure@vpos{b}}
156 \newenvironment{SCfigure}{\SC@float[\SC@figure@vpos]{figure}}{\endSC@float}
157 \newenvironment{SCfigure*}{\SC@dblfloat[\SC@figure@vpos]{figure}}{\endSC@dblfloat}
158

```

SCtable (*env.*) Simply passes the first (optional) parameter and the required one, in this case ‘table’, to **SC@float**. The table caption should be top aligned.

```

159
160 %\newcommand*{SCtableVpos{t}}% 2002/06/14
161 \@ifdefinable{SC@table@vpos}{\def\SC@table@vpos{t}}
162 \newenvironment{SCtable}{\SC@float[\SC@table@vpos]{table}}{\endSC@float}
163 \newenvironment{SCtable*}{\SC@dblfloat[\SC@table@vpos]{table}}{\endSC@dblfloat}
164

```

wide (*env.*) This is an environment that allows to extend the width of the text body (or of a floating environment) by using the margin space. It shouldn’t be used in twocolumn text.

```

165 %
166 \newenvironment{wide}%
167 {%

```

```

168 \@parboxrestore% 2002/06/14 -- SW-Arch.: wozu? 2002/08/24
169 \advance\linewidth\marginparwidth
170 \advance\linewidth\marginparsep
171 \ifx\label\SC@label
172 (+debug) \typeout{WIDE: inside SCfloat}%
173 \SC@widetrue
174 \else
175 (+debug) \typeout{WIDE: outside SCfloat}%
176 \begin{lrbox}\SC@BOX % \begin necessary: --> \@noitemerr
177 \minipage\linewidth
178 \fi
179 }%
180 {%
181 \ifx\label\SC@label
182 (+debug)
183 \typeout{WIDE: inside SCfloat}%
184 %\renewcommand*\{isSC@WIDEi}[2]{\if@twocolumn ##2\else ##1\fi}%
185 %\renewcommand*\{isSC@WIDEii}[2]{##1}%
186 %\let\isSC@WIDE\isSC@WIDEi
187 \else
188 (+debug)
189 \typeout{WIDE: outside SCfloat}%
190 \endminipage
191 \end{lrbox}% \end necessary: --> \@noitemerr
192 \global\advance\c@SC@C\@ne
193 \SC@orig@label{\SC@IDENT}% nicht \ref... !!!
194 \trivlist \item\relax
195 \isSC@ODD{\def\SC@hpos{1}}{\def\SC@hpos{r}}%
196 (+debug)
197 \typeout{WIDE: \string\linewidth=\the\linewidth}%
198 \noindent\makebox[\textwidth][\SC@hpos]{\copy\SC@BOX}% \usebox{\SC@BOX}
199 \endtrivlist
200 \fi
201 }
202

```

5.5 Internal macros

5.5.1 Collecting arguments

The new internal float environment, similar/analogous to L^AT_EX's `@float` environment.

Syntax: `\SC@float[<vpos>]{<name>}[<relwidth>][<fps>]`

- Parameter *<vpos>* (optional) is the vertical positioning of the caption.
- Parameter *<name>* (required) is the name of the 'original' L^AT_EX floating environment (e.g., 'figure' or 'table').
- Parameter *<relwidth>* (optional) is the desired relative width of the caption.
- Parameter *<fps>* (optional) is the usual L^AT_EX float positioning specifier.

The usual 'cascading' programming style is applied (cf. L^AT_EX's `\@float`).

`\SC@float` Initially, the first optional parameter is checked for.

```
203 \def\SC@float{\SC@SaveCommands\ifnextchar[\SC@xfloat{\SC@xfloat[c]}}
```

`\SC@xfloat` Then the first and second parameters are consumed and the third one is checked for.

```
204 \def\SC@xfloat[#1]#2{\ifnextchar[%
205   {\SC@yfloat{#1}{#2}}%
206   {\SC@zfloat{#1}{#2}{\sidecaptionrelwidth}[\@nameuse{fps@#2}]}}
```

`\SC@yfloat` Again, the fourth (and last) parameter is checked for.

```
207 \def\SC@yfloat#1#2[#3]{\ifnextchar[%
208   {\SC@zfloat{#1}{#2}{#3}}%
209   {\SC@zfloat{#1}{#2}{#3}[\@nameuse{fps@#2}]}}
```

5.5.2 Capturing the float's contents

`\SC@zfloat` Finally, here is the macro that does all the work.

```
210 \def\SC@zfloat#1#2#3[#4]{%
211   \def\SC@vpos{#1}%
212   \expandafter\edef\csname fps@#2\endcsname{#4}%
213   \def\SC@captype{#2}%
214   \ifx#3@empty\def\SC@fraction{1}\else\def\SC@fraction{#3}\fi%
```

The `\caption` and `\label` commands must be redefined.

```
215 \let\SC@CAPtext@empty \let\SC@OPTCAPtext@empty \let\SC@LABtext@empty%
216 \renewcommand\caption[2][\gdef\SC@OPTCAPtext{##1}%
217 \gdef\SC@CAPtext{##2}\ignorespaces}% \SC@justify
218 \let\label=\SC@label
219 \@namedef{fnum@#2}{\mbox{\@nameuse{#2name}~\@nameuse{the#2}}}% caption2 ?
```

Save the figure or table (or whatever) in a box.

```
220 \SC@widefalse
221 \begin{lrbox}\SC@BOX % \begin necessary: --> \@noitemerr
222   %%% \strut
223 }%
224
```

`\SC@dblfloat` Analogous to L^AT_EX's `\@dblfloat`.

```
225 \def\SC@dblfloat{%
226   \SC@SaveCommands%
227   \if@twocolumn\let\reserved@a\SC@dbflt\else\let\reserved@a\SC@float\fi
228   \reserved@a}
229 \def\SC@dbflt{\SC@float}
```

5.5.3 Output the float's contents

`\endSC@float` Outputs the figure or table (or whatever) and the caption.

```
230 \def\endSC@float{%
231   %%% \strut
232   \end{lrbox}% \end necessary: --> \@noitemerr
233 (+debug) \typeout{onecolumn}%
234 \let\isSC@WIDE\isSC@WIDEi%
235 \def\@FLOAT{\@float}\def\end@FLOAT{\end@float}%
```

```

236 \isSC@WIDE%
237   {\@tempdima\textwidth % \columnwidth ???
238   \advance\@tempdima\marginparwidth
239   \advance\@tempdima\marginparsep}%
240   {\@tempdima\columnwidth}%
241 (+debug) \typeout{endSC@float: \string\@tempdima=\the\@tempdima}%
242 \endSC@FLOAT\@tempdima}%

```

`\endSC@dblfloat` Ditto for *-forms of floats.

```

243 \def\endSC@dblfloat{%
244   \end{lrbox}% \end necessary: --> \@noitemerr
245 (+debug) \typeout{twocolumn}%
246 \let\isSC@WIDE\isSC@WIDEii%
247 \def\@FLOAT{\@dblfloat}\def\end@FLOAT{\end@dblfloat}%
248 \isSC@WIDE%
249   {\@tempdima\textwidth%
250   \advance\@tempdima\marginparwidth
251   \advance\@tempdima\marginparsep}%
252   {\@tempdima\textwidth}%
253 (+debug)
254 \typeout{endSC@dblfloat: \string\@tempdima=\the\@tempdima}%
255 \endSC@FLOAT\@tempdima}%

```

`\endSC@FLOAT` Sets the caption width. If caption width plus figure/table width (plus separation space) is too large then the caption width is set equal to the remaining width.

```

256 \def\endSC@FLOAT#1{%
257   \SC@calcwidths{#1}%

```

`\@FLOAT` Calls the L^AT_EX float command with the two minipages inside a main minipage.

```

258 \@FLOAT{\SC@capttype}%
259   \abovecaptionskip\z@skip
260   \belowcaptionskip\z@skip

```

Creates a label for each figure or table (etc.) for later determination if the page is odd or even. The counter `SC@C` must be incremented before.

```

261 \global\advance\c@SC@C\@ne
262 \SC@orig@label{\SC@IDENT}% nicht \ref... !!!

```

`\isSC@WIDE`

```

263 \isSC@WIDE%
264   {\ifdim\SC@tempdimb>\textwidth % ???
265   \isSC@ODD{\def\SC@hpos{1}}{\def\SC@hpos{r}}}%
266   \else%
267   \def\SC@hpos{c}%
268   \fi
269   \@tempdimc\textwidth}% ???
270   {\@tempdimc\SC@tempdima\def\SC@hpos{c}}}%
271 \makebox[\@tempdimc][\SC@hpos]{%

```

`\SC@FLOAT` Has two parameters. The first parameter is the minipage with the caption text inside and the last parameter is the minipage with the body of the figure or table inside.


```

272 \SC@FLOAT%
273   {\minipage[\SC@vpos]{\SC@CAPWD}%
274     %% \nointerlineskip %\offinterlineskip%
275     %% \kernOpt\relax
276     \hrule \@height\z@\@depth\z@  %% Testen, Doku. (Implementierung) !!!
277     %\show\SC@justify@caption
278 %%% \tracingall\tracingonline=0
279 \SC@justify
280 \let\@arrayparboxrestore\relax % sonst Blocksatz! (\@parboxrestore)
281 \let\usecaptionmargin\relax % sonst Pseudo-Blocksatz, nicht-alternierend!
282 % \isSC@ODD {\RaggedRight }{\RaggedLeft }
283 %%% A B C
284 % \isSC@ODD{\typeout{xxxxx ungerade}}{\typeout{xxxxx gerade}}%
285 % \def\@makecaption##1##2{##1: ##2\endgraf}%
286 % \@parboxrestore
287 % \@setminipage
288 \iffalse
289 % DEBUG:
290   \endgraf
291   \@makecaption{\cname fnum@\SC@captype\endcname}%
292     {\ignorespaces \SC@CAPtext}%
293   \endgraf
294 \else
295   % \ifthenelse{\equal{\SC@OPTCAPtext}{\@empty}}%
296   \ifx\SC@OPTCAPtext\@empty% Noetig?
297 % FM: strut needed twice for multiline captions
298   \SC@orig@caption[\SC@CAPtext]%
299     {\strut\ignorespaces\SC@CAPtext\unskip\strut}% \expandafter\protect
300   \else
301 % FM: strut needed twice for multiline captions
302 % FM: if \SC@CAPtext can contain trailing banks then we also need \unskip
303 % FM: (not checked)
304   \SC@orig@caption[\SC@OPTCAPtext]% \expandafter\protect
305     {\strut\ignorespaces\SC@CAPtext\unskip\strut}% \expandafter\protect
306   \fi
307   \unskip% ??? Dok. !!!
308   \ifx\SC@LABtext\@empty\else
309     \SC@orig@label{\SC@LABtext}\fi% \expandafter\protect
310 \fi
311 %%% X Y Z
312   \hrule \@height\z@\@depth\z@
313   \endminipage}%
314 {\minipage[\SC@vpos]{\SC@BOXWD}%
315   %% \nointerlineskip %\offinterlineskip%
316   %% \kernOpt\relax
317   \hrule \@height\z@\@depth\z@
318   \SC@justify@body
319   \unhbox\SC@BOX % \strut \unhbox/\usebox{\SC@BOX}% TESTEN!
320   % \kernOpt\relax
321   \hrule \@height\z@\@depth\z@
322   \endminipage}%
323 }%
324 \end@FLOAT%
325 }

```

```
326 \InputIfFileExists{sidecap.cfg}{-}{-} % hier ? % 2002/06/14
327 </package>
```

Change History

v1.0 (RN)		v1.5f (RN)	
General: New	1	General: New option ‘raggedleft’	4
v1.1 (RN/HjG)		v1.5g (RN)	
General: First public version	1	General: New option ‘ragged’	4
v1.2 (HjG)		v1.5h (RN/HjG)	
General: Proposal for amendments	1	General: Environment wide does	
v1.3 (RN/HjG)		the right thing now outside	
General: Unsuccessful attempt to		SCfloat environments.	1
make star-forms work	1	Improved compatibility with	
v1.4c (HjG)		package hyperref.	1
General: Successful (?) attempt to		\SC@SaveCommands:	
make star-forms work, plus		Renamed \SC@AtFirstCall	3
some amendments	1	v1.6 (RN)	
v1.4d (RN)		General: “Smart” loading of	
General: New (LPPL) license	1	package ragged2e. More simple	
v1.5a (RN)		code for hyperref compatibility	
General: New Option and		(\hypergetpageref).	5
environment ‘wide’; some small		v1.6 (RN/HjG)	
changes	4	General: New option	
v1.5b (RN)		‘margincaption’ added.	4
General: New Option ‘raggedright’		Oneside mode had been	
as suggested by James Kilfiger		forgotten ;-)	3
(mailto:mapdn@csv.warwick.ac.uk)		v1.6d (RN/HjG)	
.....	4	General: Simplify internal	
v1.5d (RN/HjG)		calculations using primitive	
General: Compatibility with		operators where possible and	
‘hyperref’, thanks to Heiko		useful, saving tokens (suggested	
Oberdiek	3	by Frank Mittelbach.	1
Slight amendments to		v1.6f (RN/HjG)	
documentation.	1	General: Test for presence of	
v1.5e (RN)		package babel (\org@pageref).	5
\SC@SaveCommands:		v1.7a (RN)	
\SC@AtFirstCall: Saves the old		General: Special handling with	
meaning of \caption and		\SC@pageref is no longer	
\label at first call of some SC		necessary.	5
macros (compatibility with			
‘hyperref’).	3		

Index

Numbers written in *italic* refer to the page where the corresponding entry is described, the ones underlined to the definition, the rest to the places where the entry is used.

Symbols	\@arrayparboxrestore	280
\@FLOAT	235, 247, <u>258</u> , 258	

- `\@dblfloat` 247
`\@depth` 276, 312, 317, 321
`\@empty` 99,
 100, 101, 214,
 215, 295, 296, 308
`\@float` 235
`\@height`
 . 276, 312, 317, 321
`\@ifdefinable`
 5, 6, 8, 9, 155, 161
`\@ifnextchar`
 203, 204, 207
`\@ifpackageloaded` . 136
`\@ifundefined` 146
`\@makecaption` . 285, 291
`\@namedef` 18, 219
`\@nameuse` . 206, 209, 219
`\@ne` 192, 261
`\@noitemerr` ... 176,
 191, 221, 232, 244
`\@parboxrestore` ...
 168, 280, 286
`\@setminipage` 287
`\@tempdima`
 . 31, 32, 33, 34,
 35, 38, 237, 238,
 239, 240, 241,
 242, 249, 250,
 251, 252, 254, 255
`\@tempdimc` 269, 270, 271
- A**
- `\abovecaptionskip` . 259
`\advance` 32, 33, 35, 36,
 105, 106, 169,
 170, 192, 238,
 239, 250, 251, 261
`\AtBeginDocument` .. 134
- B**
- `\begin` 176, 221
`\belowcaptionskip` . 260
- C**
- `\c@SC@C` 192, 261
`\caption` 54, 216
`\Centering` ... 121, 150
`\centering` 150
`\columnwidth` .. 237, 240
`\copy` 198
`\csname` 212, 291
- D**
- `\DeclareOption`
 ... 63, 70, 77,
- 82, 86, 89, 91,
 93, 96, 97, 110,
 115, 120, 124, 128
- E**
- `\edef` 15, 212
`\else` . 43, 44, 60, 87,
 135, 174, 184,
 187, 214, 227,
 266, 294, 300, 308
`\end` 191, 232, 244
`\end@dblfloat` 247
`\end@FLOAT` 235, 247, 324
`\end@float` 235
`\endcsname` ... 212, 291
`\endgraf` .. 285, 290, 293
`\endminipage`
 190, 313, 322
`\endSC@dblfloat` ...
 157, 163, 243
`\endSC@FLOAT`
 242, 255, 256
`\endSC@float`
 156, 162, 230
`\endtrivlist` 199
- environments:
- `SCfigure` 153
`SCtable` 159
`wide` 165
`\equal` 295
`\ExecuteOptions` ... 131
`\expandafter` .. 212,
 299, 304, 305, 309
- F**
- `\fi` 37, 43, 44, 62, 87,
 145, 178, 184,
- 200, 214, 227,
 268, 306, 309, 310
- G**
- `\gdef` 12, 216, 217
`\global` . 22, 23, 192, 261
- H**
- `\hrule` 276, 312, 317, 321
`\hspace` .. 65, 72, 78, 83
- I**
- `\if@twocolumn`
 87, 184, 227
`\if@twoside` 57
`\ifdim` 34, 264
`\iffalse` 22, 288
`\IfFileExists` 138
`\ifSC@origRagged` .. 135
`\ifSC@wide` 22, 23, 43, 44
`\ifthenelse` 59, 295
`\iftrue` 23
`\ifx` 171,
 181, 214, 296, 308
`\ignorespaces` .. 12,
 217, 292, 299, 305
`\InputIfFileExists` .
 132, 326
`\isodd` 59
`\isSC@First` 46
`\isSC@ODD`
 . 58, 61, 65, 67,
 72, 74, 112, 117,
 195, 265, 282, 284
`\isSC@WIDE`
 . 186, 234, 236,
 246, 248, 263, 263
`\isSC@WIDEi` 41,
 43, 87, 184, 186, 234
`\isSC@WIDEii`
 42, 44, 88, 185, 246
`\item` 194
- K**
- `\kern` 275, 316, 320
- L**
- `\label` . 54, 171, 181, 218
`\let` 22, 23, 54, 94, 99,
 100, 101, 148,
 149, 150, 186,
 215, 218, 227,
 234, 246, 280, 281
`\linewidth`
 . 169, 170, 177, 197

- M**
- `\makebox` 198, 271
 - `\marginparsep` . . 15, 106, 170, 239, 251
 - `\marginparwidth` 105, 107, 169, 238, 250
 - `\mbox` 219
 - `\minipage` . 177, 273, 314
- N**
- `\newboolean` 21, 52
 - `\newcommand` 10, 11, 12, 13, 14, 16, 17, 27, 41, 42, 43, 44, 46, 48, 49, 50, 53, 58, 61, 154, 160
 - `\newcounter` 10
 - `\newdimen` 5, 6, 8, 9
 - `\newenvironment` 156, 157, 162, 163, 166
 - `\newsavebox` 10
 - `\noindent` 198
 - `\nointerlineskip` 274, 315
 - `\number` 11
- O**
- `\offinterlineskip` 274, 315
- P**
- `\PackageWarningNoLine` 141
 - `\pageref` 59
 - `\ProcessOptions` . . . 133
 - `\protect` 299, 304, 305, 309
- R**
- `\RaggedLeft` 67, 74, 84, 92, 112, 117, 125, 148, 282
 - `\raggedleft` 148
 - `\RaggedRight` . . . 67, 74, 79, 90, 112, 117, 129, 149, 282
 - `\raggedright` 149
 - `\ref` 193, 262
 - `\relax` 28, 103, 133, 194, 275, 280, 281, 316, 320
- S**
- `\renewcommand` 64, 66, 71, 73, 78, 83, 87, 88, 184, 185, 216
 - `\RequirePackage` . 3, 139
 - `\reserved@a` 227, 228
- S**
- `\SC@BOX` 10, 29, 176, 198, 221, 319
 - `\SC@BOXWD` 5, 29, 30, 31, 104, 105, 106, 314
 - `\SC@calcwidths` 27, 102, 257
 - `\SC@CAPtext` 215, 217, 292, 298, 299, 302, 305
 - `\SC@capttype` 213, 258, 291
 - `\SC@CAPWD` 6, 30, 32, 35, 36, 107, 273
 - `\SC@dbflt` 227, 229
 - `\SC@dblfloat` 157, 163, 225
 - `\SC@figure@vpos` 155, 156, 157
 - `\SC@FLOAT` 10, 64, 71, 78, 83, 272, 272
 - `\SC@float` 156, 162, 203, 227, 229
 - `\SC@fraction` . . . 30, 214
 - `\SC@hpos` 195, 198, 265, 267, 270, 271
 - `\SC@IDENT` 11, 59, 193, 262
 - `\SC@justify` 48, 90, 92, 94, 217, 279
 - `\SC@justify@body` 50, 111, 116, 121, 125, 129, 318
 - `\SC@justify@caption` 49, 66, 73, 79, 84, 94, 277
 - `\SC@label` 12, 171, 181, 218
 - `\SC@LABtext` 12, 215, 308, 309
 - `\SC@OPTCAPtext` 215, 216, 295, 296, 304
 - `\SC@orig@caption` 13, 54, 298, 304
 - `\SC@orig@label` 13, 54, 193, 262, 309
 - `\SC@SaveCommands` 53, 56, 203, 226
 - `\SC@table@vpos` 161, 162, 163
 - `\SC@tempdima` 8, 28, 34, 36, 270
 - `\SC@tempdimb` 9, 38, 103, 104, 264
 - `\SC@vpos` 211, 273, 314
 - `\SC@widefalse` . . . 22, 220
 - `\SC@widetrue` 23, 173
 - `\SC@xfloat` 203, 204
 - `\SC@yfloat` 205, 207
 - `\SC@zfloat` 206, 208, 209, 210
 - `SCfigure (env.)` 153
 - `\SCfigureVpos` 154
 - `SCtable (env.)` 159
 - `\SCtableVpos` 160
 - `\setboolean` 52, 96
 - `\show` 277
 - `\sidecaptionrelwidth` 16, 206
 - `\sidecaptionsep` 14, 15, 33, 65, 72, 78, 83
 - `\sidecaptionvpos` . . 17
 - `\string` 197, 241, 254
 - `\strut` 222, 231, 299, 305, 319
- T**
- `\textwidth` 198, 237, 249, 252, 264, 269
 - `\the` 15, 197, 241, 254
 - `\tracingall` 278
 - `\tracingonline` 278
 - `\trivlist` 194
 - `\typeout` 172, 175, 183, 189, 197, 233, 241, 245, 254, 284
- U**
- `\unhbox` 319
 - `\unskip` 299, 302, 305, 307
 - `\usebox` 198, 319
 - `\usecaptionmargin` . 281
- V**
- `\value` 11
- W**
- `\wd` 29
 - `wide (env.)` 165
- Z**
- `\z@` 276, 312, 317, 321
 - `\z@skip` 259, 260