

# The Implementation of the caption package<sup>\*</sup>

Axel Sommerfeldt

[axel.sommerfeldt@f-m.fm](mailto:axel.sommerfeldt@f-m.fm)

2011/08/06

## Abstract

The caption package consists of two parts – the kernel (`caption3.sty`) and the main package (`caption.sty`).

The caption package redefines the  $\LaTeX$  commands `\caption`, `\@caption`, and `\@makecaption` and maps the latter one to `\caption@@make`, giving the user the possibility to control the look & feel of the captions from floating environments like `figure` and `table`. Furthermore it does similar to the caption stuff coming from other packages (like the `longtable` or `supertabular` package): Mapping the appropriate internal commands (like `\LT@makecaption` or `\ST@caption`) to the ones offered by the `caption3` kernel. So you can think of the caption package as a layer package, it simply provides adaptation layers between the caption stuff coming from  $\LaTeX 2_{\epsilon}$  or packages, and the caption stuff offered by the `caption3` kernel.

## User manuals

This document is describing the code implementation only. The user documentation can be found in

<a href="#">caption-eng.pdf</a>	The English documentation
<a href="#">caption-rus.pdf</a>	The Russian documentation <sup>1</sup>
<a href="#">caption-deu.pdf</a>	The German documentation

---

<sup>\*</sup>This package has version number v3.2, last revised 2011/08/06.

<sup>1</sup>Thanks a lot to Olga Lapko for this translation

# Contents

<b>1 Identification</b>	<b>4</b>
<b>2 Loading the kernel</b>	<b>4</b>
<b>3 Check against incompatible document classes</b>	<b>4</b>
<b>4 Check against incompatible packages</b>	<b>4</b>
<b>5 Declaration of options</b>	<b>4</b>
5.1 Options for figure and table . . . . .	4
5.2 Miscellaneous options . . . . .	5
5.3 caption v1.x compatibility options . . . . .	5
5.4 caption2 v2.x compatibility options . . . . .	6
5.5 Obsolete caption v3.0 options . . . . .	6
5.6 fltpage package support options . . . . .	7
5.7 hyperref package support options . . . . .	7
<b>6 <math>\mathcal{A}\mathcal{M}\mathcal{S}</math> &amp; SMF document classes support</b>	<b>7</b>
<b>7 KOMA-Script document classes support</b>	<b>7</b>
<b>8 Processing of options</b>	<b>9</b>
<b>9 <code>\caption</code>, <code>\@caption</code>, and <code>\@makecaption</code></b>	<b>9</b>
<b>10 <code>\captionof</code> and <code>\captionlistentry</code></b>	<b>18</b>
<b>11 <code>\captionbox</code></b>	<b>21</b>
<b>12 <code>\ContinuedFloat</code></b>	<b>22</b>
<b>13 Internal helpers</b>	<b>23</b>
<b>14 Support for sub-captions</b>	<b>26</b>
<b>15 Document class &amp; Babel package support</b>	<b>28</b>
15.1 The $\mathcal{A}\mathcal{M}\mathcal{S}$ & SMF classes . . . . .	28
15.2 The beamer class . . . . .	28
15.3 The KOMA-Script classes . . . . .	29
15.4 The frenchb Babel option . . . . .	29
15.5 The frenchle/pro package . . . . .	29
15.6 The hungarian and magyar Babel option . . . . .	30

<b>16 Package support</b>	<b>30</b>
16.1 The float package . . . . .	32
16.2 The floatflt package . . . . .	35
16.3 The fltpage package . . . . .	36
16.4 The hyperref package . . . . .	38
16.5 The hypcap package . . . . .	42
16.6 The listings package . . . . .	42
16.7 The longtable package . . . . .	43
16.8 The picinpar package . . . . .	46
16.9 The picins package . . . . .	47
16.10 The rotating package . . . . .	49
16.11 The sidecap package . . . . .	50
16.12 The subfigure package . . . . .	51
16.13 The supertabular and xtab packages . . . . .	52
16.14 The threeparttable package . . . . .	53
16.15 The wrapfig package . . . . .	54

## 1 Identification

```
1 \NeedsTeXFormat{LaTeX2e}[1994/12/01]
2 \ProvidesPackage{caption}[2011/08/06 v3.2 Customizing captions (AR)]
3 %\@ifundefined{PackageRedefines}{}{\PackageRedefines{caption}{caption}}
```

## 2 Loading the kernel

```
4 \RequirePackage{caption3}[2011/08/01] % needs v1.3 or newer
```

## 3 Check against incompatible document classes

```
5 \caption@ifbool{documentclass}{}{%
6   \caption@WarningNoLine{%
7     Unsupported document class (or package) detected, \MessageBreak
8     usage of the caption package is not recommended}%
9   \caption@InfoNoLine{\string\@makecaption\space=\space\meaning\@makecaption}%
10 }
```

## 4 Check against incompatible packages

```
11 \@ifpackageloaded{caption2}{%
12   \caption@Error{%
13     You can't use both, the (obsolete) caption2 *and* \MessageBreak
14     the (current) caption package}%
15   \endinput
16 }{}

17 \caption@AtBeginDocument{%
18   \@ifpackageloaded{ftcap}{\caption@DisablePositionOption{ftcap}}{}%
19   \@ifpackageloaded{nonfloat}{\caption@DisablePositionOption{nonfloat}}{}%
20   \@ifpackageloaded{topcapt}{\caption@DisablePositionOption{topcapt}}{}%

\caption@DisablePositionOption \caption@DisablePositionOption{<package>}
disables the 'position' option.

21 \newcommand*\caption@DisablePositionOption[1]{%
22   \caption@InfoNoLine{%
23     '#1' package detected; setting 'position=b' for compatibility reasons}%
24   \caption@setposition b%

25   \DeclareCaptionOption{position}{%
26     \caption@Error{Usage of the 'position' option is incompatible \MessageBreak
27       to the '#1' package}}%

28 \@onlypreamble\caption@DisablePositionOption
```

## 5 Declaration of options

### 5.1 Options for figure and table

```
29 \DeclareCaptionOption{figureposition}{%
30   \captionsetup*[figure]{position=#1}}
31 \@onlypreamble@key{caption}{figureposition}

32 \DeclareCaptionOption{tableposition}{%
33   \captionsetup*[table]{position=#1}}
```

```

34 \@onlypreamble@key{caption}{tableposition}
35 \DeclareCaptionOption{figurename}{\caption@SetName{figure}{#1}}
36 \DeclareCaptionOption{tablename}{\caption@SetName{table}{#1}}
37 \DeclareCaptionOption{name}{\caption@setname\@capttype{#1}}
38 \DeclareCaptionOption{listfigurename}{\caption@SetName{listfigure}{#1}}
39 \DeclareCaptionOption{listtablename}{\caption@SetName{listtable}{#1}}

\caption@SetName \caption@SetName{<cmd>}{<value>}
40 \newcommand*\caption@SetName[2]{%
41   \caption@setname{#1}{#2}%
42   \begin{group}
43     \ifundefined{language}{\language}{%
44       \ifundefined{captions\language}{\language}{%
45         \expandafter\g@addto@macro\csname captions\language\endcsname
46           {\caption@setname{#1}{#2}}}%
47     \end{group}}
48 \newcommand*\caption@setname[2]{\@namedef{#1name}{#2}}
49 \caption@AtBeginDocument{\let\caption@SetName\caption@setname}

\caption@DeclareWithinOption
50 \newcommand*\caption@DeclareWithinOption[1]{%
51   \DeclareCaptionOption{#1within}{\caption@Within{#1}{#1}}%
52   \DeclareCaptionOption{#1without}{\caption@Within{#1}{none}}%

53 \caption@DeclareWithinOption{figure}
54 \caption@DeclareWithinOption{table}

55 \DeclareCaptionOption{within}{%
56   \ifundefined{c@figure}{\caption@Within{figure}{#1}}%
57   \ifundefined{c@table}{\caption@Within{table}{#1}}%
58   \caption@For{typelist}{\caption@Within{##1}{#1}}%
59   \def\caption@within@default{#1}}
60 \DeclareCaptionOption{without}{\KV@caption@within{none}}

\caption@within
61 \newcommand*\caption@Within[1]{\def\caption@type{#1}\KV@caption@DCT@within}

```

## 5.2 Miscellaneous options

```

62 \DeclareCaptionOption*{config}[caption]{%
63   \InputIfFileExists{#1.cfg}%
64   {\typeout{*** Local configuration file #1.cfg used ***}}%
65   {\caption@Warning{Configuration file #1.cfg not found}}}

66 \DeclareCaptionOption{@minipage}{%
67   \caption@ifinlist{#1}{auto,default}%
68   {\let\caption@if@minipage\@gobbles}%
69   {\caption@set@bool\caption@if@minipage{#1}}}
70 \caption@setup{@minipage=default}

```

## 5.3 caption v1.x compatibility options

```

71 \DeclareCaptionOption{compatibility}[1]{\caption@setbool{compatibility}{#1}}
72 \@onlypreamble@key{caption}{compatibility}

```

```

73 \DeclareCaptionOptionNoValue*{normal}{%
74   \caption@setformat{plain}%
75   \caption@setjustification{justified}}
76 \DeclareCaptionOptionNoValue*{isu}{%
77   \caption@setformat{hang}%
78   \caption@setjustification{justified}}
79 \DeclareCaptionOptionNoValue*{hang}{%
80   \caption@setformat{hang}%
81   \caption@setjustification{justified}}
82 \DeclareCaptionOptionNoValue*{center}{%
83   \caption@setformat{plain}%
84   \caption@setjustification{centering}}
85 \DeclareCaptionOptionNoValue*{anne}{%
86   \caption@setformat{plain}%
87   \caption@setjustification{centerlast}}
88 \DeclareCaptionOptionNoValue*{centerlast}{%
89   \caption@setformat{plain}%
90   \caption@setjustification{centerlast}}

91 \DeclareCaptionOptionNoValue*{scriptsize}{\def\captionfont{\scriptsize}}
92 \DeclareCaptionOptionNoValue*{footnotesize}{\def\captionfont{\footnotesize}}
93 \DeclareCaptionOptionNoValue*{small}{\def\captionfont{\small}}
94 \DeclareCaptionOptionNoValue*{normalsize}{\def\captionfont{\normalsize}}
95 \DeclareCaptionOptionNoValue*{large}{\def\captionfont{\large}}
96 \DeclareCaptionOptionNoValue*{Large}{\def\captionfont{\Large}}

97 \DeclareCaptionOptionNoValue*{up}{\l@addto@macro\captionlabelfont\upshape}
98 \DeclareCaptionOptionNoValue*{it}{\l@addto@macro\captionlabelfont\itshape}
99 \DeclareCaptionOptionNoValue*{sl}{\l@addto@macro\captionlabelfont\slshape}
100 \DeclareCaptionOptionNoValue*{sc}{\l@addto@macro\captionlabelfont\scshape}
101 \DeclareCaptionOptionNoValue*{md}{\l@addto@macro\captionlabelfont\mdseries}
102 \DeclareCaptionOptionNoValue*{bf}{\l@addto@macro\captionlabelfont\bfseries}
103 \DeclareCaptionOptionNoValue*{rm}{\l@addto@macro\captionlabelfont\rmfamily}
104 \DeclareCaptionOptionNoValue*{sf}{\l@addto@macro\captionlabelfont\sffamily}
105 \DeclareCaptionOptionNoValue*{tt}{\l@addto@macro\captionlabelfont\ttfamily}

106 \DeclareCaptionOptionNoValue*{nooneline}{\caption@setbool{slc}{0}}
107 \caption@setbool{ruled}{0}
108 \DeclareCaptionOptionNoValue*{ruled}{\caption@setbool{ruled}{1}}

```

## 5.4 caption2 v2.x compatibility options

```

109 \DeclareCaptionOptionNoValue*{flushleft}{%
110   \caption@setformat{plain}%
111   \caption@setjustification{raggedright}}
112 \DeclareCaptionOptionNoValue*{flushright}{%
113   \caption@setformat{plain}%
114   \caption@setjustification{raggedleft}}

115 \DeclareCaptionOptionNoValue*{oneline}{\caption@setbool{slc}{1}}

116 \DeclareCaptionOptionNoValue*{ignoreLTcapwidth}{%
117   \caption@WarningNoLine{Obsolete option 'ignoreLTcapwidth' ignored}}

```

## 5.5 Obsolete caption v3.0 options

```

118 \DeclareCaptionOption*{caption}{%
119   \caption@setbool{temp}{#1}%

```

```

120 \caption@ifbool{temp}{\}%
121 \caption@Error{%
122     The package option 'caption=#1' is obsolete.\MessageBreak
123     Please pass this option to the subfig package instead\MessageBreak
124     and do *not* load the caption package anymore}}

```

## 5.6 fltpage package support options

With these options is controlled where the list-of entry and `\ref` resp. `\pageref` or `\autoref` will link to. Defaults are `FList=caption` and `FPref=figure` which is inconsistent, but compatible to the usual behaviour of the `fltpage` package.

```

125 \DeclareCaptionOption{FList}[1]{\caption@setFPoption{list}{#1}}
126 \DeclareCaptionOption{FPref}[1]{\caption@setFPoption{ref}{#1}}
127 \@onlypreamble@key{caption}{FList}
128 \@onlypreamble@key{caption}{FPref}

129 \newcommand*\caption@setFPoption[2]{%
130     \edef\caption@tempa{\@car#2\@nil}%
131     \caption@setbool{FP#1cap}{\if c\caption@tempa 1\else 0\fi}}
132 \@onlypreamble\caption@setFPoption
133 \captionsetup{FList=caption,FPref=figure}

```

## 5.7 hyperref package support options

With `hypcap=off` one can turn the `hypcap` support off (default is on).

```

134 \DeclareCaptionOption{hypcap}[1]{\caption@setbool{hypcap}{#1}}
135 \DeclareCaptionOption{hypcapSPACE}{\def\caption@hypcapSPACE{#1}}
136 \captionsetup{hypcap=1,hypcapSPACE=.5\baselineskip}

```

## 6 $\mathcal{A}\mathcal{M}\mathcal{S}$ & SMF document classes support

```

137 \caption@ifamsclass{%
138     \caption@InfoNoLine{AMS or SMF document class}%
139     \setlength\belowcaptionskip{0pt}% set to 12pt by AMS class
140 }

```

## 7 KOMA-Script document classes support

```

141 \caption@ifkomaclass{%
142     \caption@InfoNoLine{KOMA-Script document class}%

```

Here we emulate the caption related commands and take over the caption related settings from the KOMA-Script classes.

```

\@tablecaptionabovetrue
\@tablecaptionabovefalse
143 \g@addto@macro\@tablecaptionabovetrue{\captionsetup*[table]{position=t}}
144 \g@addto@macro\@tablecaptionabovefalse{\captionsetup*[table]{position=b}}

145 \if@tablecaptionabove
146     \@tablecaptionabovetrue
147 \else
148     \@tablecaptionabovefalse
149 \fi

```

```

\onelinecaptionstrue
\onelinecaptionsfalse 150 \g@addto@macro\onelinecaptionstrue{\let\caption@ifslc\@firstoftwo}
151 \g@addto@macro\onelinecaptionsfalse{\let\caption@ifslc\@secondoftwo}

152 \ifonelinecaptions
153 \onelinecaptionstrue
154 \else
155 \onelinecaptionsfalse
156 \fi

\@captionabovetrue Please note that these are stronger than the position setting, therefore we override the
\@captionabovefalse options figureposition and tableposition to typeout a warning.

157 \g@addto@macro\@captionabovetrue{\let\caption@position\@firstoftwo}
158 \g@addto@macro\@captionabovefalse{\let\caption@position\@secondoftwo}

159 \DeclareCaptionOption{figureposition}{%
160 \caption@WarningNoLine{Option 'figureposition=#1' has no effect\MessageBreak
161 when used with a KOMA script document class}}
162 \DeclareCaptionOption{tableposition}{%
163 \caption@WarningNoLine{Option 'tableposition=#1' has no effect\MessageBreak
164 when used with a KOMA script document class}}

\setcapindent

165 \let\caption@KOMA@setcapindent\@setcapindent
166 \renewcommand*\@setcapindent[1]{%
167 \caption@KOMA@setcapindent{#1}\caption@setcapindent}

168 \let\caption@KOMA@@setcapindent\@setcapindent
169 \renewcommand*\@@setcapindent[1]{%
170 \caption@KOMA@@setcapindent{#1}\caption@setcapindent}

171 \newcommand*\caption@setcapindent{%
172 \captionsetup{indent=\ifdim\cap@indent<\z@\z@\else\cap@indent\fi}}

173 \@ifundefined{cap@indent}{}{\caption@setcapindent}

\setcapwidth Note: The optional argument of \setcapwidth if not supported (yet), so we issue a warning if
used. (Since this does not seem to have an negative effect when used by the captionbeside
environment, we suppress the warning here.)

174 \expandafter\let\expandafter\caption@KOMA@setcapwidth
175 \csname\string\setcapwidth\endcsname
176 \@namedef{\string\setcapwidth}[#1]#2{%
177 \caption@KOMA@setcapwidth[#1]{#2}\caption@setcapwidth{#1}}

178 \newcommand*\caption@setcapwidth[1]{%
179 \ifx\#1\else
180 \@ifundefined{cap@margin}{}{%
181 \def\@tempa{captionbeside}%
182 \ifx\@tempa\@currenvir\else\caption@Warning{%
183 Ignoring optional argument [1] of \string\setcapwidth\MessageBreak}%
184 \fi}%
185 \fi
186 \captionsetup{width=\cap@width}}

```



```

187 \def\caption@tempa{\hsize}%
188 \ifx\caption@tempa\cap@width \else
189   \caption@setcapwidth{?}
190 \fi

\setcapmargin
191 \expandafter\let\expandafter\caption@KOMA@setcapmargin
192   \csname\string\@setcapmargin\endcsname
193 \@namedef{\string\@setcapmargin}[#1]#2{%
194   \caption@KOMA@setcapmargin[#1]{#2}\caption@setcapmargin}
195 \expandafter\let\expandafter\caption@KOMA@@setcapmargin
196   \csname\string\@@setcapmargin\endcsname
197 \@namedef{\string\@@setcapmargin}[#1]#2{%
198   \caption@KOMA@@setcapmargin[#1]{#2}\caption@setcapmargin}
199 \newcommand*\caption@setcapmargin{%
200   \begingroup
201   \let\onelinecaptionsfalse\relax
202   \def\@twoside{0}%
203   \def\if@twoside{\def\@twoside{1}\iffalse}%
204   \cap@margin
205   \def\@tempa{\endgroup}%
206   \ifx\cap@left\hfill\else\ifx\cap@right\hfill\else
207     \def\hspace##1##{\@firstofone}%
208     \edef\@tempa{\endgroup
209       \noexpand\captionsetup{%
210         twoside=\@twoside,slc=0,%
211         margin={\cap@left,\cap@right}}}%
212   \fi\fi
213   \@tempa}

214 \ifx\cap@margin\relax \else
215   \caption@setcapmargin
216 \fi
217 }

```

## 8 Processing of options

```

218 \caption@SetupOptions{caption}{\caption@setkeys{#1}{#2}}%
219 \caption@ProcessOptions*{caption}

```

## 9 \caption, \@caption, and \@makecaption

\caption@caption Here comes our definition of \caption and \caption\*. Beside the support of the starred variant this code was adapted to the various packages we support. We are using \caption@dblarg instead of \@dblarg so \caption{} (with an empty arg.) will produce a list-of entry, but \caption[]{} won't.

```

220 \def\caption@caption{%
221   \caption@iftype
222     {\caption@checkgrouplevel\@empty\caption
223     \caption@star
224     {\caption@refstepcounter\@captype}}%

```

```

225         {\caption@dblarg{\@caption\@capttype}}}%
226         {\caption@Error{\noexpand\caption outside float}}}%

```

`\caption@star` A helper macro which processes the optional `*` after `\caption`.

*Note:* We set `\caption@startrue` globally so it works with the `sidecap` package, too.

```

227 \newcommand*\caption@star[2]{%
228   \@ifstar{\global\caption@startrue#2[]}{#1#2}}%

```

`\caption@@caption` As above, our version has been adapted to the packages we support. Additionally our code is nested by `\caption@beginex` & `\caption@end` instead of `\begingroup` & `\endgroup`. Furthermore we use `\caption@boxrestore` instead of `\@parboxrestore` so this code also works correctly inside list-based environments like `wide` & `addmargin`. (This, and the fact that we use `\linewidth` instead of `\hsize` inside `\@makecaption`, solves [L<sup>A</sup>T<sub>E</sub>X PR latex/2472](#).)

```

229 \long\def\caption@@caption#1[#2]#3{%
230   \ifcaption@star \else
231     \caption@prepareanchor{#1}{#2}%
232   \fi
233   \par
234   \caption@beginex{#1}{#2}{#3}%
235   \caption@setfloatcapt{%
236     \caption@boxrestore
237     \if@minipage
238       \@setminipage
239     \fi
240     \caption@normalsize
241     \ifcaption@star
242       \let\caption@makeanchor\@firstofone
243     \fi
244     \@makecaption{\csname fnum@#1\endcsname}%
245                   {\ignorespaces\caption@makeanchor{#3}}\par
246     \caption@if@minipage\@minipagetrue\@minipagefalse}%
247   \caption@end}%

```

`\caption@prepareanchor`

```

248 \newcommand*\caption@prepareanchor[2]{%
249   \caption@makecurrent{#1}{#2}%
250   \caption@ifhypcap\caption@@start{}}

```

`\caption@makecaption` `\@makecaption{<label>}{<text>}`

We do basically the same as the original code (from the standard `LATEX` document classes), but take care of the `position=` setting and use `\caption@@make` from the `caption` kernel to finally typeset the caption.

```

251 \long\def\caption@makecaption#1#2{%
252   \caption@iftop
253     {\vskip\belowcaptionskip}%
254     {\caption@rule\vskip\abovecaptionskip}%
255   \caption@@make{#1}{#2}%
256   \caption@iftop
257     {\vskip\abovecaptionskip\caption@rule}%
258     {\vskip\belowcaptionskip}}

```

```

\caption@redefine We only redefine \caption and \@caption if the current definitions are well known,
so documents written in the old (caption package v1.x) days (where \caption &
\@caption were not redefined by us) will still compile fine. For example the usage
of the captcont package, which brings it's own definition of \caption*, was quite
common these days.

259 \newcommand*\caption@redefine{}
260 \g@addto@macro\caption@redefine{%

261   \caption@setbool{incompatible}{0}%

262   \caption@CheckCommand\caption{%
263     % ltfloat.dtx [2002/10/01 v1.1v LaTeX Kernel (Floats)]
264     \def\caption{%
265       \ifx\@capttype\@undefined
266         \@latex@error{\noexpand\caption outside float}\@ehd
267         \expandafter\@gobble
268       \else
269         \refstepcounter\@capttype
270         \expandafter\@firstofone
271       \fi
272       {\@dblarg{\@caption\@capttype}}%
273     }%

274   \caption@CheckCommand\caption{%
275     % beamerbaselocalstructure.sty,v 1.53 2007/01/28 20:48:21 tantau
276     \def\caption{
277       \ifx\@capttype\@undefined
278         \@latex@error{\noexpand\caption outside figure or table}\@ehd
279         \expandafter\@gobble
280       \else
281         \refstepcounter\@capttype
282         \expandafter\@firstofone
283       \fi
284       {\@dblarg{\@caption\@capttype}}%
285     }%

286   \caption@CheckCommand\caption{%
287     % float.sty [2001/11/08 v1.3d Float enhancements (AL)]
288     \renewcommand\caption{%
289       \ifx\@capttype\@undefined
290         \@latex@error{\noexpand\caption outside float}\@ehd
291         \expandafter\@gobble
292       \else
293         \refstepcounter\@capttype
294         \let\@tempf\@caption
295         \expandafter\ifx\csname @float@c@\@capttype\endcsname\relax\else
296           \expandafter\expandafter\let
297           \expandafter\@tempf\csname @float@c@\@capttype\endcsname
298         \fi
299         \fi
300         \@dblarg{\@tempf\@capttype}}}%

301   \caption@CheckCommand\caption{%
302     % hyperref.sty [2007/02/27 v6.75t Hypertext links for LaTeX]
303     % hyperref.sty [2007/04/09 v6.76a Hypertext links for LaTeX]
304     % hyperref.sty [2007/06/12 v6.76h Hypertext links for LaTeX]

```

```

305 \def\caption{%
306   \ifx\@capttype\@undefined
307     \latex@error{\noexpand\caption outside float}\@ehd
308     \expandafter\@gobble
309   \else
310     \H@refstepcounter\@capttype
311     \@ifundefined{fst@\@capttype}{%
312       \let\Hy@tempa\@caption
313     }{%
314       \let\Hy@tempa\Hy@float@caption
315     }%
316     \expandafter\@firstofone
317   \fi
318   {\@dblarg{\Hy@tempa\@capttype}}%
319 }%

320 \caption@CheckCommand\caption{%
321   % hyperref.sty [2007/08/05 v6.76j Hypertext links for LaTeX]
322   \def\caption{%
323     \ifx\@capttype\@undefined
324       \latex@error{\noexpand\caption outside float}\@ehd
325       \expandafter\@gobble
326     \else
327       \H@refstepcounter\@capttype
328       \let\Hy@tempa\@caption
329       \@ifundefined{float@caption}{%
330         }{%
331           \expandafter\ifx\csname @float@c@\@capttype\endcsname\float@caption
332             \let\Hy@tempa\Hy@float@caption
333           \fi
334         }%
335       \expandafter\@firstofone
336     \fi
337     {\@dblarg{\Hy@tempa\@capttype}}%
338   }%

339 \caption@IfCheckCommand{}{%
340   \caption@InfoNoLine{%
341     Incompatible package detected (regarding \string\caption).\MessageBreak
342     \string\caption\space=\space meaning\caption}%
343   \caption@setbool{incompatible}{1}}%

344 \caption@CheckCommand\@caption{%
345   % ltfloat.dtx [2002/10/01 v1.1v LaTeX Kernel (Floats)]
346   \long\def\@caption#1[#2]#3{%
347     \par
348     \addcontentsline{\csname ext@#1\endcsname}{#1}%
349     {\protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}}%
350     \begingroup
351       \@parboxrestore
352       \if@minipage
353         \setminipage
354       \fi
355       \normalsize
356       \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
357     \endgroup}%

```

```

358 \caption@CheckCommand\@caption{%
359 % beamerbaselocalstructure.sty,v 1.53 2007/01/28 20:48:21 tantau
360 \long\def\@caption#1[#2]#3{% second argument ignored
361 \par\nobreak
362 \begingroup
363 \@parboxrestore
364 \if@minipage
365 \setminipage
366 \fi
367 \beamer@makecaption{#1}{\ignorespaces #3}\par\nobreak
368 \endgroup}}%

369 \caption@CheckCommand\@caption{%
370 % magyar.ldf [2005/03/30 v1.4j Magyar support from the babel system]
371 \long\def\@caption#1[#2]#3{%
372 \csname par\endcsname
373 \addcontentsline{\csname ext@#1\endcsname}{#1}%
374 {\protect\numberline{\csname the#1\endcsname.}{\ignorespaces #2}}%
375 \begingroup
376 \@parboxrestore
377 \if@minipage
378 \setminipage
379 \fi
380 \normalsize
381 \makecaption{\csname fnum@#1\endcsname}%
382 {\ignorespaces #3}\csname par\endcsname
383 \endgroup}}%

384 % \caption@CheckCommand\float@caption{%
385 % float.sty [2001/11/08 v1.3d Float enhancements (AL)]
386 \long\def\float@caption#1[#2]#3{%
387 \addcontentsline{\@nameuse{ext@#1}}{#1}%
388 {\protect\numberline{\@nameuse{the#1}}{\ignorespaces #2}}
389 \global\setbox\@floatcapt\vbox\bgroup\@parboxrestore
390 \normalsize\@fs@capt{\@nameuse{fnum@#1}}{\ignorespaces #3}%
391 \@ifnextchar[{\float@ccon}{\egroup}}%
392 \long\def\float@ccon[#1]{#1\par\egroup}}%

393 \caption@CheckCommand\@caption{%
394 % hyperref.sty [2007/02/27 v6.75t Hypertext links for LaTeX]
395 \long\def\@caption#1[#2]#3{%
396 \hyper@makecurrent{\@cuptype}%
397 \def\@currentlabelname{#2}%
398 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
399 \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}}%
400 }%
401 \begingroup
402 \@parboxrestore
403 \if@minipage
404 \setminipage
405 \fi
406 \normalsize
407 \makecaption{\csname fnum@#1\endcsname}{%
408 \ignorespaces
409 \ifHy@nesting
410 \hyper@@anchor{\@currentHref}{#3}%

```

```

411         \else
412             \Hy@raisedlink{\hyper@@anchor{\@currentHref}{\relax}}#3%
413         \fi
414     }%
415     \par
416 \endgroup
417 } }%

418 \caption@CheckCommand\@caption{%
419 % hyperref.sty [2007/04/09 v6.76a Hypertext links for LaTeX]
420 % hyperref.sty [2007/06/12 v6.76h Hypertext links for LaTeX]
421 % hyperref.sty [2007/08/05 v6.76j Hypertext links for LaTeX]
422 \long\def\@caption#1[#2]#3{%
423     \expandafter\ifx\csname if@capstart\expandafter\endcsname
424         \csname iftrue\endcsname
425     \global\let\@currentHref\hc@currentHref
426 \else
427     \hyper@makecurrent{\@cuptype}%
428 \fi
429 \def\@currentlabelname{#2}%
430 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
431     \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
432 }%
433 \begingroup
434     \@parboxrestore
435     \if@minipage
436         \@setminipage
437     \fi
438     \normalsize
439     \expandafter\ifx\csname if@capstart\expandafter\endcsname
440         \csname iftrue\endcsname
441         \global\@capstartfalse
442         \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
443     \else
444         \@makecaption{\csname fnum@#1\endcsname}{%
445             \ignorespaces
446             \ifHy@nesting
447                 \hyper@@anchor{\@currentHref}{#3}%
448             \else
449                 \Hy@raisedlink{\hyper@@anchor{\@currentHref}{\relax}}#3%
450             \fi
451         }%
452     \fi
453 \par
454 \endgroup
455 } }%

456 \caption@CheckCommand\@caption{%
457 % hyperref.sty [2009/11/27 v6.79k Hypertext links for LaTeX]
458 \long\def\@caption#1[#2]#3{%
459     \expandafter\ifx\csname if@capstart\expandafter\endcsname
460         \csname iftrue\endcsname
461     \global\let\@currentHref\hc@currentHref
462 \else
463     \hyper@makecurrent{\@cuptype}%

```

```

464 \fi
465 \def\@currentlabelname{#2}%
466 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
467 \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
468 }%
469 \begingroup
470 \@parboxrestore
471 \if@minipage
472 \setminipage
473 \fi
474 \normalsize
475 \expandafter\ifx\csname if@capstart\expandafter\endcsname
476 \csname iftrue\endcsname
477 \global\@capstartfalse
478 \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
479 \else
480 \@makecaption{\csname fnum@#1\endcsname}{%
481 \ignorespaces
482 \ifHy@nesting
483 \expandafter\hyper@anchor\expandafter{\@currentHref}{#3}%
484 \else
485 \Hy@raisedlink{%
486 \expandafter\hyper@anchor\expandafter{\@currentHref}{\relax}%
487 }%
488 #3%
489 \fi
490 }%
491 \fi
492 \par
493 \endgroup
494 }%

495 \caption@CheckCommand\@caption{%
496 % hyperref.sty [2009/12/09 v6.79m Hypertext links for LaTeX]
497 % hyperref.sty [2009/12/28 v6.79z Hypertext links for LaTeX]
498 \long\def\@caption#1[#2]#3{%
499 \expandafter\ifx\csname if@capstart\expandafter\endcsname
500 \csname iftrue\endcsname
501 \global\let\@currentHref\hc@currentHref
502 \else
503 \hyper@makecurrent{\@capttype}%
504 \fi
505 \@ifundefined{NR@getttitle}{%
506 \def\@currentlabelname{#2}%
507 }{%
508 \NR@getttitle{#2}%
509 }%
510 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
511 \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
512 }%
513 \begingroup
514 \@parboxrestore
515 \if@minipage
516 \setminipage
517 \fi

```

```

518         \normalsize
519         \expandafter\ifx\csname if@capstart\expandafter\endcsname
520             \csname iftrue\endcsname
521             \global\@capstartfalse
522             \makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
523         \else
524             \makecaption{\csname fnum@#1\endcsname}{%
525                 \ignorespaces
526                 \ifHy@nesting
527                     \expandafter\hyper@@anchor\expandafter{\@currentHref}{#3}%
528                 \else
529                     \Hy@raisedlink{%
530                         \expandafter\hyper@@anchor\expandafter{%
531                             \@currentHref
532                             }{\relax}%
533                     }%
534                 #3%
535             \fi
536         }%
537     \fi
538     \par
539 \endgroup
540 }%

541 \caption@CheckCommand\@caption{%
542     % nameref.sty [2006/12/27 v2.28 Cross-referencing by name of section]
543     \long\def\@caption#1[#2]{%
544         \def\@currentlabelname{#2}%
545         \NR@@caption{#1}{{#2}}%
546     }%

547 \caption@CheckCommand\@caption{%
548     % nameref.sty [2009/11/27 v2.32 Cross-referencing by name of section]
549     \long\def\@caption#1[#2]{%
550         \NR@getttitle{#2}%
551         \NR@@caption{#1}{{#2}}%
552     }%

553 \caption@CheckCommand\@caption{%
554     % subfigure.sty [2002/07/30 v2.1.4 subfigure package]
555     \long\def\@caption#1[#2]#3{%
556         \ifundefined{if#1topcap}%
557             {\subfig@oldcaption{#1}{{#2}}{#3}}%
558             {\@nameuse{if#1topcap}%
559                 \@listsubcaptions{#1}%
560                 \subfig@oldcaption{#1}{{#2}}{#3}%
561             \else
562                 \subfig@oldcaption{#1}{{#2}}{#3}%
563                 \@listsubcaptions{#1}%
564             \fi}}}%

565 \caption@CheckCommand\@caption{%
566     % subfig.sty [2005/06/28 ver: 1.3 subfig package]
567     \def\@caption{\caption@}%
568     \long\def\caption@#1[#2]#3{%
569         \ifundefined{caption@setfloattype}%
570         \caption@settype

```



```

571 %         \caption@setfloattype
572 %         \@capttype
573 %         \sf@ifpositiontop{%
574 %             \@listsubcaptions{#1}%
575 %             \sf@old@caption{#1}[\{#2\}]{#3}%
576 %         }{%
577 %             \sf@old@caption{#1}[\{#2\}]{#3}%
578 %             \@listsubcaptions{#1}%
579 %         }}%
580     }%

581 \caption@ifcheckcommand{}{%
582     \caption@ifnonline{%
583         Incompatible package detected (regarding \string\@caption).\MessageBreak
584         \string\@caption\space=\space\meaning\@caption}%
585     \caption@setbool{incompatible}{1}}%

```

The option `compatibility=` will override the compatibility mode.

```

586 \@ifundefined{caption@ifcompatibility}%
587 {
588     \let\caption@ifcompatibility\caption@ifincompatible
589     \let\caption@tempa\caption@WarningNoLine}%
590 {
591     \let\caption@tempa\@gobble}% suppress warning
592
593 \caption@ifcompatibility{%
594     \caption@tempa{%
595         \noexpand\caption will not be redefined since it's already\MessageBreak
596         redefined by a document class or package which is\MessageBreak
597         unknown to the caption package}%
598     \renewcommand*\caption@redefine{}%

```

`\ContinuedFloat` is not supported in compatibility mode.

```

596 \renewcommand*\caption@ContinuedFloat[1]{%
597     \caption@Error{Not available in compatibility mode}}%

```

`\caption@start` is not supported in compatibility mode.

```

598 \caption@AtBeginDocument*{%
599     \let\caption@start\relax
600     \@ifundefined{caption@ORI@capstart}{}{%
601         \caption@Debug{%
602             Restore hypcap definition of \string\capstart\@gobble}%
603         \let\capstart\caption@ORI@capstart}%
604     \@ifundefined{caption@ORI@float@makebox}{}{%
605         \caption@Debug{%
606             Restore hyperref redefinition of \string\float@makebox\@gobble}%
607         \let\float@makebox\caption@ORI@float@makebox}%
608     }%

```

`\caption@star` We redefine `\caption@star` here so it does not make any harm.

```

609 \renewcommand*\caption@star[2]{#1#2}%
610 }{%
611 \caption@ifincompatible{%
612     \caption@WarningNoLine{%
613         Forced redefinition of \noexpand\caption since the\MessageBreak
614         unsupported(!) package option 'compatibility=false'\MessageBreak

```

```

615         was given}%
616     }{}%

\caption
\@caption 617     \renewcommand*\caption@redefine{%
618         \let\caption\caption@caption
619         \let\@caption\caption@@caption}%
620     \caption@redefine

621 }%

622 \caption@AtBeginDocument*{%
623     \let\caption@ORI@capstart\@undefined
624     \let\caption@ORI@float@makebox\@undefined}%

\@xfloat We redefine \@xfloat so inside floating environments our type-specific options will be
used, a hyperref anchor will be set etc.

625 \let\caption@ORI@xfloat\@xfloat
626 \def\@xfloat#1[#2]{%
627     \caption@ORI@xfloat{#1}[#2]%
628     \caption@settype{#1}}%

629 }

```

Some packages (like the `hyperref` package for example) redefines `\caption` and `\@caption`, too. So we have to use `\AtBeginDocument` here, so we can make sure our definition is the one which will be valid at last.

```

630 \caption@AtBeginDocument{\caption@redefine}

\@makecaption
631 \let\@makecaption\caption@makecaption

```

## 10 `\captionof` and `\captionlistentry`

```

632 \caption@AtBeginDocument{%
633     \DeclareCaptionOption{type}{\setcaptiontype{#1}}%
634     \DeclareCaptionOption{type*}{\setcaptiontype*{#1}}%
635     \DeclareCaptionOption{subtype}[sub\@capytype]{\caption@setsubtype{#1}}%
636     \DeclareCaptionOption{subtype*}[sub\@capytype]{\caption@setsubtype*{#1}}%
637 }

```

*Important Note:* Like `\captionof` the option `type=` should only be used inside a group, box, or environment and does not check if the argument is a valid floating environment or not.

```

\setcaptiontype Like \captionsetup{type=xxx}, but also works if \captionsetup was redefined.

638 \newcommand\setcaptiontype{%
639     \caption@boxrestore@mini
640     \caption@settype}

```

`\caption@settype` `\caption@settype*{<type>}`  
 sets `\@capttype` and executes the options associated with it (using `\caption@setoptions`).  
 Furthermore we check `\currentgrouplevel` (if avail), redefine `\@currentlabel`  
 so a `\label` before `\caption` will result in a hint instead of a wrong reference, and  
 use the macro `\caption@(sub)typehook` (which will be used by our float package  
 support).

The non-starred version sets a `hyperref` anchor additionally (if `hypcap=true` and the  
`hypcap` package is not loaded).

```

641 \newcommand*\caption@settype{%
642   \caption@@settype{}}

643 \newcommand*\caption@setsubtype{%
644   \caption@iftype
645     {\caption@settype{sub}}%
646     {\caption@Error{Option 'subtype=' outside float}}}%

647 \newcommand*\caption@@settype[1]{%
648   \caption@teststar{\caption@@@settype{#1}}{\@firstoftwo\@secondoftwo}

649 \newcommand*\caption@@@settype[3]{%
650   % #1 = "" or "sub"
651   % #2 = \@firstoftwo in star form, \@secondoftwo otherwise
652   % #3 = <type>, e.g. "figure" or "table"
653   \caption@Debug{#1type=#3}%
654   \caption@checkgrouplevel{#1}{%
655     \captionsetup{#1type#2*\@empty=...}#2{ or
656       \@backslashchar#1captionof}{}}%

657   \edef\caption@tempa{#3}%
658   \expandafter\ifx\csname @#1capttype\endcsname\caption@tempa \else
659     \ifcaptionsetup@star\else\@nameuse{caption@#1type@warning}\fi
660   \fi
661   \expandafter\let\csname @#1capttype\endcsname\caption@tempa
662   \@nameuse{caption@#1typehook}%

663   \caption@setoptions{#3}%
664   \ifx\caption@opt\relax
665     \@nameundef{caption@#1type@warning}%
666   \else
667     \@namedef{caption@#1type@warning}{\caption@Warning{%
668       The #1caption type was already set to
669       '\csname @#1capttype\endcsname'\MessageBreak}}%
670   \fi

671   \let\caption@ifrefstepcounter\@secondoftwo
672   #2{ }{%
673     \let\@currentlabel\caption@undefinedlabel
674   %   \let\@currentHlabel\@undefined
675     \ifx\caption@ORI@label\@undefined
676       \let\caption@ORI@label\label
677     \let\label\caption@xlabel
678   \fi
679   \caption@start}}

```

`\caption@typehook` Hook, will be extended later on, e.g. by our float package support.

```

680 \newcommand*\caption@typehook{ }

```

`\caption@iftype` Since we often need to check if `\@capttype` is defined (means: we are inside a floating environment) this helper macro was introduced.

```

681 \newcommand*\caption@iftype{%
682   \@ifundefined{@capttype}{\let\@capttype\@undefined\@secondoftwo}\@firstoftwo}

```

`\caption@checkgrouplevel` Checks if `\captionsetup{type=...}` or `\caption` is done inside a group or not – in the latter case a warning message will be issued. (needs  $\epsilon$ -TeX)

```

683 \begingroup\expandafter\expandafter\expandafter\endgroup
684 \expandafter\ifx\csname currentgrouplevel\endcsname\relax
685   \caption@Debug{TeX engine: TeX}
686   \let\caption@checkgrouplevel\@gobbletwo
687 \else
688   \caption@Debug{TeX engine: e-TeX}
689   \newcommand*\caption@checkgrouplevel[2]{%
690     \@ifundefined{#1caption@grouplevel}{%
691       \@ifundefined{caption@grouplevel}{\let\caption@grouplevel\z@}{}}%
692     \ifnum\currentgrouplevel>\caption@grouplevel\relax
693       \expandafter\edef\csname #1caption@grouplevel\endcsname{%
694         \the\currentgrouplevel}%
695     \else
696       \caption@Warning{\string#2\MessageBreak outside box or environment}%
697     \fi
698   }{}}
699 \fi

```

`\caption@undefinedlabel` This label will be used for `\currentlabel` inside (floating) environments as default. (see above)

```

700 \newcommand*\caption@undefinedlabel{%
701   \protect\caption@xref{\caption@labelname}{\on@line}}
702 \DeclareRobustCommand*\caption@xref[2]{%
703   \caption@WarningNoLine{\noexpand\label before \string\caption#2}%
704   \@setref\relax\@undefined{#1}}
705 \newcommand*\caption@labelname{??}

```

`\caption@xlabel` The new code of `\label` inside floating environments. `\label` will be redefined using `\caption@withoptargs`, so #1 are the optional arguments (if any), and #2 is the mandatory argument here.

```

706 \newcommand*\caption@xlabel[1]{%
707   \caption@@xlabel
708   \def\caption@labelname{#1}%
709   \caption@ORI@label{#1}}
710 \newcommand*\caption@@xlabel{%
711   \global\let\caption@@xlabel\@empty
712   \bsphack
713   \protected@write\@auxout{}%
714     {\string\providecommand*\string\caption@xref[2]{%
715       \string\@setref\string\relax\string\@undefined{\string#1}}}%
716   \esphack}

```

`\captionof` `\captionof{<type>}[<lst.entry>]{<heading>}`  
`\captionof*{<lst.entry>}{<heading>}`

*Note:* This will be defined with `\AtBeginDocument` so `\usepackage{caption,capt-of}` will still work. (Compatibility to *v1.x*)

```

717 \caption@AtBeginDocument{%
718   \def\captionof{\caption@teststar\caption@of{\caption*}\caption}}
719 \newcommand*\caption@of[2]{\setcaptiontype*{#2}#1}

\captionlistentry \captionlistentry[\float type]{\list entry}
\captionlistentry*[ \float type ]{\list entry}
720 \newcommand*\captionlistentry{%
721   \caption@teststar\@captionlistentry\@firstoftwo\@secondoftwo}

722 \newcommand*\@captionlistentry[1]{%
723   \@testopt{\caption@listentry{#1}}\@capttype}

724 \def\caption@listentry#1[#2]#3{%
725   \@bsphack
726     #1{\caption@getttitle{#3}}%
727     {\caption@refstepcounter{#2}}%
728     \caption@makecurrent{#2}{#3}}%
729   \caption@addcontentsline{#2}{#3}%
730   \@esphack}

```

## 11 `\captionbox`

`\captionbox` A `\parbox` with contents and caption, separated by an invisible `\hrule`.

```

731 \newcommand*\captionbox{%
732   \let\captionbox@type\@empty
733   \caption@withoptargs\caption@box}

734 \newcommand\caption@box[2]{%
735   \@testopt{\caption@ibox{#1}{#2}}{\wd\@tempboxa}}

736 \long\def\caption@ibox#1#2[#3]{%
737   \@testopt{\caption@iibox{#1}{#2}{#3}}\captionbox@hj@default}

738 \long\def\caption@iibox#1#2#3[#4]#5{%
739   \setbox\@tempboxa\hbox{#5}%
740   \begingroup
741     \expandafter\captionsetup\expandafter{\captionbox@type*}% set \caption@position
742     \caption@iftop{%
743       \endgroup
744       \parbox[t]{#3}{%
745         \expandafter\captionsetup\expandafter{\captionbox@type,position=t}%
746         \vbox{\caption#1{#2}}%
747         \captionbox@hrule
748         \csname caption@hj@#4\endcsname
749         \unhbox\@tempboxa}%
750       }{%
751         \endgroup
752         \parbox[b]{#3}{%
753           \expandafter\captionsetup\expandafter{\captionbox@type,position=b}%
754           \csname caption@hj@#4\endcsname
755           \unhbox\@tempboxa
756           \captionbox@hrule

```

```

757      \vtop{\caption#1{#2}}}%
758  }}

759 \newcommand*\captionbox@hj@default{c}
760 \newcommand*\captionbox@hrule{\hrule\@height\z@\relax}

761 \providecommand*\caption@hj@c{\centering}
762 \providecommand*\caption@hj@l{\raggedright}
763 \providecommand*\caption@hj@r{\raggedleft}
764 \providecommand*\caption@hj@s{}

765 \DeclareCaptionOption{*}[]{}

```

## 12 \ContinuedFloat

`\ContinuedFloat`    `\ContinuedFloat`  
`\ContinuedFloat*`

This mainly decrements the appropriate counter and increments the continuation counter instead. Furthermore we set `\caption@resetContinuedFloat` to `\@gobble` so the continuation counter will not be reset to zero inside `\caption@refstepcounter`. Please forget about the optional argument, it was never working well, is incompatible to the subfig package, but is still there for compatibility reasons.

*Note:* The definition of `\ContinuedFloat` itself is compatible to the one inside the subfig package, except for the starred variant and the optional argument.

When the `hyperref` package is used we have the problem that the usage of `\ContinuedFloat` will create duplicate hyper links – `\@currentHref` will be the same for the main float and the continued ones. So we have to make sure unique labels and references will be created each time. We do this by extending `\theHfigure` and `\theHtable`, so for continued floats the scheme

$$\langle type \rangle . \langle type \# \rangle \backslash \alpha \{ \langle continued \# \rangle \}$$

will be used instead of

$$\langle type \rangle . \langle type \# \rangle \quad .$$

(This implementation follows an idea from Steven Douglas Cochran.)

*Note:* This does not help if the `hyperref` package option `naturalnames=true` is set.

```

766 \def\ContinuedFloat{%
767   \ifnextchar[\@Continued@Float\@ContinuedFloat}
768 \def\@Continued@Float[#1]{\addtocounter{#1}{\m@ne}

769 \def\@ContinuedFloat{%
770   \caption@iftyp
771     {\addtocounter{\@capytype}{\m@ne}
772     \caption@ContinuedFloat{\@capytype}%
773     {\caption@Error{\noexpand\ContinuedFloat outside float}}}}

774 \def\caption@ContinuedFloat#1{%
775   \ifstar{\caption@Continued@Float@{#1}}{\caption@Continued@Float{#1}}}

776 \def\caption@Continued@Float@{%
777   \addtocounter{\@capytype}{\@ne}
778   \@stpelt{ContinuedFloat}\stepcounter{ContinuedFloat}%
779   \def\caption@resetContinuedFloat##1{\xdef\caption@Cftype{##1}}%
780   \caption@@ContinuedFloat}

```

```

781 \def\caption@Continued@Float#1{%
782   \edef\caption@tempa{#1}%
783   \ifx\caption@tempa\caption@CFtype
784     \stepcounter{ContinuedFloat}%
785     \let\caption@resetContinuedFloat\@gobble
786     \caption@@ContinuedFloat{#1}%
787     \sf@ContinuedFloat{#1}%
788   \else
789     \caption@Error{Continued `#1' after `\caption@CFtype'}%
790   \fi}

791 \def\caption@@ContinuedFloat#1{%
792   \expandafter\l@addto@macro\csname the#1\endcsname\theContinuedFloat
793   \@ifundefined{theH#1}{}{%
794     \expandafter\l@addto@macro\csname theH#1\endcsname{%
795       \@alph@c@ContinuedFloat}}%
796   \caption@setoptions{ContinuedFloat}%
797   \caption@setoptions{continued#1}}

798 \providecommand*\sf@ContinuedFloat[1]{}

799 \newcommand*\caption@CFtype{??}

```

`\theContinuedFloat` Its preset to `\@empty`, so usually the continuation counter is not included in the caption label or references.

```

800 \newcounter{ContinuedFloat}
801 \let\theContinuedFloat\@empty

```

`\caption@resetContinuedFloat` `\caption@resetContinuedFloat{<type>}`  
 If a continuation counter is defined, we reset it. (This one will be called inside `\@caption`.)

```

802 \newcommand*\caption@resetContinuedFloat[1]{%
803   \@stpelt{ContinuedFloat}\xdef\caption@CFtype{#1}}

```

`\phantomcaption` `\phantomcaption`  
 Use this one for figures with subcaptions but without main caption.

```

804 \newcommand\phantomcaption{%
805   \caption@iftype
806     {\caption@refstepcounter\@c@type}%
807     {\caption@Error{\noexpand\phantomcaption outside float}}}%

```

## 13 Internal helpers

`\caption@refstepcounter` Resets the continuation counter, increments the float (i.e. figure or table) counter, and sets the `refstepcounter` flag.

```

808 \newcommand*\caption@refstepcounter[1]{%
809   \@ifundefined{c@#1}%
810     {\caption@Error{No float type `#1' defined}}%
811     {\caption@resetContinuedFloat{#1}%
812       \caption@@refstepcounter{#1}%
813       \let\caption@ifrefstepcounter\@firstoftwo}}

814 \newcommand*\caption@@refstepcounter{\refstepcounter}
815 \let\caption@ifrefstepcounter\@secondoftwo

```

```

\caption@dblarg A \relax was added compared to \@dblarg so \caption{} will be expanded to
\caption[\relax]{} (and not to \caption[]{}).
816 \@ifundefined{kernel@ifnextchar}%
817   {\newcommand\caption@dblarg[1]{\@ifnextchar[{\#1}{\caption@xdblarg{\#1}}}%
818   {\newcommand\caption@dblarg[1]{\kernel@ifnextchar[{\#1}{\caption@xdblarg{\#1}}}%
819   \newcommand\caption@xdblarg[2]{\#1[{\#2\relax}]{\#2}}}%

\caption@begin Our handling of \caption will always be surrounded by \caption@begin (or
\caption@beginex) and \caption@end.
\caption@begin{<type>} performs these tasks:

    1. Start a new group.

    2. Define \fnum@<type> if the caption label format is set to non-default.

    3. Override the position= setting, if necessary. (for example if set to auto or used
       inside a supertabular)

820 \newcommand*\caption@begin[1]{%
821   \begingroup
822   \caption@setfnum{\#1}%
823   \caption@fixposition
824   \global\let\caption@fixedposition\caption@position}

\caption@beginex \caption@beginex{<type>}{<list entry>}{<heading>}
performs the same tasks as \caption@begin and additionally:

    4. Set \lst@@caption, so \fnum@lstlisting will include a numbering.

    5. Make an entry in the list-of-whatever.

    6. Set \caption@ifempty according argument <heading>.

825 \newcommand\caption@beginex[3]{%
826   \caption@begin{\#1}%
827   \let\lst@@caption\relax
828   \caption@addcontentsline{\#1}{\#2}%
829   \caption@ifempty{\#3}{}%

\caption@end \caption@end closes the group.

830 \newcommand*\caption@end{%
831   \endgroup
832   \let\caption@position\caption@fixedposition}

\caption@setfnum \caption@setfnum{<type>}
redefines \fnum@<type> according the caption label format set with labelformat=.
But if labelformat=default is set, \fnum@<type> will not be overwritten by us.

833 \newcommand*\caption@setfnum[1]{%
834   \@ifundefined{fnum@\#1}{\iftrue}{\ifx\caption@lfmt\caption@lfmt@default\else}%
835   \@namedef{fnum@\#1}{\caption@fnum{\#1}}%
836   \fi}

\caption@setparboxrestore \caption@setparboxrestore{<partial or full>}

837 \newcommand*\caption@setparboxrestore[1]{%
838   \caption@ifinlist{\#1}{full}{%

```



```

839 \let\caption@parboxrestore\@firstoftwo
840 }\caption@ifinlist{#1}{default,light,partial}}{%
841 \let\caption@parboxrestore\@secondoftwo
842 }{%
843 \caption@Error{Undefined parboxrestore `#1'}%
844 }}

```

```

tion@setfullparboxrestore \caption@setfullparboxrestore
This is an abbreviation for \caption@setparboxrestore{full}.
845 \newcommand*\caption@setfullparboxrestore{%
846 \let\caption@parboxrestore\@firstoftwo}

847 \DeclareCaptionOption{parboxrestore}{\caption@setparboxrestore{#1}}
848 \caption@setparboxrestore{default}

```

`\caption@boxrestore` **The original code (from latex/base/ltboxes.dtx):**

```

\def\@parboxrestore{\@arrayparboxrestore\let\\\@normalcr}
\def\@arrayparboxrestore{%
\let\if@nobreak\iffalse
\let\if@noskipsec\iffalse
\let\par\@par
\let\-\@dischyph
\let'\@acci\let'\@accii\let\=\@acciii
\parindent\z@ \parskip\z@skip
\everypar{}%
\linewidth\hsize
\@totalleftmargin\z@
\leftskip\z@skip \rightskip\z@skip \@rightskip\z@skip
\parfillskip\@flushglue \lineskip\normallineskip
\baselineskip\normalbaselineskip
\sloppy}

```

**This one will be used by \@caption instead of \@parboxrestore.**

```

849 \newcommand*\caption@boxrestore{%
850 \caption@parboxrestore{\@parboxrestore}{%
851 \let\if@nobreak\iffalse
852 \let\if@noskipsec\iffalse
853 \let\par\@par
854 % \let\-\@dischyph
855 % \let'\@acci\let'\@accii\let\=\@acciii
856 \parindent\z@ \parskip\z@skip
857 \everypar{}%
858 % \linewidth\hsize
859 % \@totalleftmargin\z@
860 \leftskip\z@skip \rightskip\z@skip \@rightskip\z@skip
861 \parfillskip\@flushglue \lineskip\normallineskip
862 \baselineskip\normalbaselineskip
863 \sloppy
864 \let\\\@normalcr
865 }}

```

`\caption@boxrestore@mini` **Resets \par so the very first \par in \@caption behaves quite the same as in floating environments. Will be used by \setcaptiontype.**

```

866 \newcommand\caption@boxrestore@mini{%
867   \let\par\@par
868   \parindent\z@ \parskip\z@skip
869   \sloppy}

```

`\caption@normalsize` This one will be used by `\@caption` instead of `\normalsize`.  
Its code is equivalent to

```
\caption@font{normal}%
```

but executes faster (since the starred form of `\caption@font` does not use `\setkeys` internally).

```

870 \newcommand*\caption@normalsize{%
871   \caption@font*\KV@caption@fnt@normal\@unused}}

```

`\caption@setfloatcapt` Needed for support of the float package, where the caption will not be typeset directly, but caught in a `\vbox` called `\@floatcapt` instead.

```
872 \let\caption@setfloatcapt\@firstofone
```

`\caption@makecurrent` All these are needed for support of the hyperref package.

```

\caption@makeanchor 873 \newcommand*\caption@makecurrent[2]{}
\caption@start      874 \let\caption@makeanchor\@firstofone
\caption@@start     875 \let\caption@start\relax
\caption@freezeHref 876 \let\caption@@start\relax
\caption@defrostHref 877 \let\caption@freezeHref\relax
                    878 \let\caption@defrostHref\relax

```

`\caption@getttitle` This one is needed for support of the nameref package.

```

879 \newcommand\caption@getttitle[1]{%
880   \@ifundefined{NR@getttitle}%
881   {\def\@currentlabelname{#1}}%
882   {\NR@getttitle{#1}}}

```

## 14 Support for sub-captions

`\caption@DeclareSubType` `\caption@DeclareSub` initializes the usage of `\caption` in sub-floats.

```

883 \def\caption@DeclareSubType sub#1\@nil{%
884   \caption@Debug{Initializing subtype for `#1'\@gobble}%
885   \@namedef{caption@c@#1}{0}%
886   \@namedef{caption@beginsub#1}{\caption@beginsubfloat{#1}}
887   \@onlypreamble\caption@DeclareSubType

```

Initialize the sub-captions defined with `\DeclareCaptionSubType...`

```
888 \caption@For*{subtypelist}{\caption@DeclareSubType sub#1\@nil}
```

Initialize the sub-captions defined with `\newsubfloat[18]...`

```

889 \caption@AtBeginDocument*{%
890   \@ifundefined{sf@counterlist}{}{%
891     \@for\sf@temp:=\sf@counterlist\do{%
892       \expandafter\caption@DeclareSubType\sf@temp\@nil}}}

```

`\caption@subtypehook` Hook, will be used inside `\caption@setsubtype`.

```

893 \newcommand*\caption@subtypehook{%
894   \ifx\caption\caption@subcaption \else

```

```

895 \caption@ifrefstepcounter{}{%
896 % no \caption or \subcaption in this (floating) environment yet
897 \caption@Debug{Increment \@capttype\ counter =\the\value\@capttype}%
898 \caption@l@stepcounter\@capttype
899 \let\caption@@addcontentsline\caption@addsubcontentsline}%
900 \ifnum\csname caption@c@\@capttype\endcsname=\value\@capttype \else
901 \caption@Debug{Reset sub\@capttype\ counter}%
902 \expandafter\xdef\csname caption@c@\@capttype\endcsname{%
903 \the\value\@capttype}%
904 \@stpelt\@subcapttype
905 \fi
906 \c@ContinuedFloat=0\relax
907 \let\caption@resetContinuedFloat\@gobble
908 \let\caption@addcontentsline\caption@kernel@addcontentsline
909 \let\caption@setfloatcapt\@firstofone
910 \caption@clearmargin
911 \caption@iflist{}{\let\caption@setlist\@gobble}%
912 \caption@setoptions{sub}%
913 \caption@setoptions{subfloat}% for subfig-package compatibility
914 \let\caption\caption@subcaption
915 \let\@makecaption\caption@makecaption
916 \fi}%

```

`\caption@subcaption` **Makes a sub-caption.**

```

917 \newcommand*\caption@subcaption{%
918 \caption@iftype
919 {\caption@checkgrouplevel{sub}\subcaption
920 \caption@star
921 {\caption@refstepcounter\@subcapttype}%
922 {\caption@dblarg{\@caption\@subcapttype}}}%
923 {\caption@Error{\noexpand\subcaption outside float}}}

```

`\caption@addcontentsline` **We extend `\caption@addcontentsline` so it handles sub-captions, too.**

*Note:* `\sf@ifpositiontop` & `\@listsubcaptions` are defined by the subfigure & subfig packages.

```

924 \let\caption@kernel@addcontentsline\caption@addcontentsline
925 \renewcommand*\caption@addcontentsline[2]{%
926 \sf@ifpositiontop{\@listsubcaptions{#1}}}%
927 \caption@kernel@addcontentsline{#1}{#2}%
928 \sf@ifpositiontop{}{\@listsubcaptions{#1}}}%
929 \caption@addsubcontentslines{#1}}
930 \newcommand*\caption@addsubcontentslines[1]{%
931 \begingroup
932 \caption@subcontentslines
933 \endgroup
934 \caption@clearsubcontentslines}%

```

`\caption@addsubcontentsline` **Add a pending sub-caption list entry.**

```

935 \newcommand*\caption@addsubcontentsline[4]{%
936 \begingroup
937 \let\label\@gobble \let\index\@gobble \let\glossary\@gobble

```

```

938 \protected@edef\@tempa{\endgroup
939 \noexpand\g@addto@macro\noexpand\caption@subcontentslines{%
940 \noexpand\@namedef{the#2}{\csname the#2\endcsname}%
941 \ifx\@currentHref\@undefined \else
942 \noexpand\def\noexpand\@currentHref{\@currentHref}%
943 \fi
944 \protect\caption@@addcontentsline{#1}{#2}{#3}{#4}}}%
945 \@tempa}

```

ion@checksubcontentslines Checks if the list of pending sub-captions is empty, if not, a warning will be issued.

```

946 \newcommand*\caption@checksubcontentslines{%
947 \ifx\caption@subcontentslines\@empty \else
948 \caption@Error{%
949 Something's wrong--perhaps a missing \protect\caption\MessageBreak
950 in the last figure or table}%
951 \caption@clearsubcontentslines
952 \fi}

```

ion@clearsubcontentslines Clear pending sub-caption list entries.

```

953 \newcommand*\caption@clearsubcontentslines{%
954 \global\let\caption@subcontentslines\@empty}

955 \caption@AtBeginDocument*{%
956 \ifundefined{sf@ifpositiontop}{\let\sf@ifpositiontop\@gobbletwo}{}}%
957 \caption@clearsubcontentslines
958 \g@addto@macro\caption@typehook{\caption@checksubcontentslines}%
959 \AtEndDocument{\caption@checksubcontentslines}}%

```

## 15 Document class & Babel package support

### 15.1 The $\mathcal{A}\mathcal{M}\mathcal{S}$ & SMF classes

```

960 \ifundefined{smf@makecaption}{{}\let\smf@makecaption\@makecaption}

```

### 15.2 The beamer class

```

961 \ifclassloaded{beamer}{%
962 \caption@InfoNoLine{beamer document class}%

```

\figure We redefine figure & table so our type-specific options will be used etc.

```

\table 963 \expandafter\let\expandafter\caption@ORI@figure
964 \csname\string\figure\endcsname
965 \@namedef{\string\figure}[#1]{%
966 \caption@ORI@figure[#1]%
967 \caption@settype{figure}}

968 \expandafter\let\expandafter\caption@ORI@table
969 \csname\string\table\endcsname
970 \@namedef{\string\table}[#1]{%
971 \caption@ORI@table[#1]%
972 \caption@settype{table}}

973 }{}

```

### 15.3 The KOMA-Script classes

KOMA-Script contains the code `\AtBeginDocument{\let\scr@caption\caption}` so we need to update `\scr@caption` here, too.

```
974 \@ifundefined{scr@caption}{}{%
975   \caption@AtBeginDocument{\let\scr@caption\caption}}
```

### 15.4 The frenchb Babel option

Suppress “Package frenchb. ldf Warning: The definition of `\@makecaption` has been changed, frenchb will NOT customize it.” (but only if we emulate this customization)

```
976 \@nameuse{caption@frenchb}\@nameundef{caption@frenchb}
```

### 15.5 The frenchle/pro package

```
977 \caption@AtBeginDocument{\@ifundefined{frenchTeXmods}{}{%
978   \caption@InfoNoLine{frenchle/pro package is loaded}%
979   \let\captionfont@ORI\captionfont
980   \let\captionlabelfont@ORI\captionlabelfont
981   \let\@makecaption@ORI\@makecaption}
```

If `\GOfrench` is defined as `\relax` all the re-definitions regarding captions have already been done, so we can do our patches immediately. Otherwise we must add our stuff to `\GOfrench`.

```
982 \@ifundefined{GOfrench}%
983   {\let\caption@tempa\@firstofone}%
984   {\def\caption@tempa{\g@addto@macro\GOfrench}}%
985   \caption@tempa{%
986     \let\captionfont\captionfont@ORI
987     \let\captionfont@ORI\@undefined
988     \let\captionlabelfont\captionlabelfont@ORI
989     \let\captionlabelfont@ORI\@undefined
990     \let\@makecaption\@makecaption@ORI
991     \let\@makecaption@ORI\@undefined
```

`\@cnORI` We update the definition of `\@cnORI` so it actually reflects our definition of `\caption`.

```
992   \let\@cnORI\caption
```

`\@tablescaption` The frenchle/pro package sets `\caption` to `\@tablescaption` at `\begin{table}` for special treatment of footnotes. Therefore we have to patch `\@tablescaption` so `\caption*` will work inside the table environment.

```
993   \let\caption@tcORI\@tablescaption
994   \def\@tablescaption{\caption@star\relax\caption@tcORI}%
```

`\f@ffrench` `\f@tfrench` and `\f@tffrench` reflect `\fnum@figure` and `\fnum@table` when used in French mode. These contain additional code which typesets the caption separator `\captionseparator` instead of the usual colon. Because this breaks with our `\@makecaption` code we have to remove this additional code here.

```
995   \let\@eatDP\@undefined
996   \let\caption@tempa\@empty
997   \ifx\f@ffrench\fnum@figure
```

```

998     \l@addto@macro\caption@tempa{\let\fnun@figure\ffrench}%
999     \fi
1000    \ifx\ffrench\fnun@table
1001    \l@addto@macro\caption@tempa{\let\fnun@table\ffrench}%
1002    \fi
1003    \def\ffrench{\ifx\listoffigures\relax\else\figurename~\thefigure\fi}%
1004    \def\ffrench{\ifx\listoftables\relax\else\tablename~\thetable\fi}%
1005    \caption@tempa

1006  }%
1007 }}

```

## 15.6 The hungarian and magyar Babel option

```

1008 \def\caption@tempa#1{%
1009   \@ifundefined{extras#1}\caption@AtBeginDocument\@firstofone{%
1010     \@ifundefined{extras#1}{}{%
1011       \caption@InfoNoLine{#1 babel option is loaded}%
1012       \expandafter\addto\csname extras#1\endcsname{%
1013         % reverse changes made by magyar.ldf
1014         \let\makecaption\caption@makecaption
1015         \babel@save\makecaption
1016         \caption@redefine
1017         \babel@save\caption}%
1018       }}}
1019 \caption@tempa{hungarian}%
1020 \caption@tempa{magyar}%

```

## 16 Package support

```

\caption@ifpackageloaded \caption@ifpackageloaded{<package>}[<version>]{<true>}{<false>}
Some kind of combination of \ifpackageloaded and \ifpackagelater. If
the <package> is not loaded yet, the check will be (re-)done \AtBeginDocument, so
the <package> could be loaded later on, too.

1021 \newcommand\caption@ifpackageloaded[1]{%
1022   \@testopt{\caption@ifpackageloaded{#1}}{}%
1023   \@onlypreamble\caption@ifpackageloaded

1024 \long\def\caption@ifpackageloaded#1[#2]#3#4{%
1025   \@ifpackageloaded{#1}\@firstofone{%
1026     \caption@Debug{#1 package is not loaded (yet)\@gobble}%
1027     \caption@AtBeginDocument{%
1028       \caption@ifpackageloaded{#1}[#2][#3][#4]}%
1029   \@onlypreamble\caption@ifpackageloaded

1030 \newcommand\caption@ifpackageloaded[1]{%
1031   \@testopt{\caption@ifpackageloaded{#1}}{}%
1032   \@onlypreamble\caption@ifpackageloaded

1033 \long\def\caption@ifpackageloaded#1[#2]{%
1034   \@ifpackageloaded{#1}{%
1035     \caption@InfoNoLine{#1 package is loaded}%
1036     \@ifpackagelater{#1}[#2]\@firstoftwo{%
1037       \caption@Error{%
1038         For a successful cooperation we need at least version\MessageBreak

```

```

1039         '#2' of package #1,\MessageBreak
1040         but only version\MessageBreak
1041         '\csname ver@#1.\@pkgextension\endcsname'\MessageBreak
1042         is available}%
1043     \@secondoftwo}%
1044 }{\@secondoftwo}}
1045 \@onlypreamble\caption@ifpackageloaded

```

`\caption@clearmargin` This macro will be used by some package support stuff where the usual margin setting is not welcome, e.g. in the `sidecap` package.

```

1046 \newcommand*\caption@clearmargin{%
1047   \setcaptionmargin\z@
1048   \let\caption@minmargin\@undefined}

1049 \caption@setbool{needfreeze}{0}
1050 \caption@AtBeginDocument*{%
1051   \caption@ifneedfreeze{%

```

`\caption@freeze` `\caption@freeze*`  
Used by the `fitpage` & `sidecap` package support.

```

1052 \newcommand*\caption@freeze{%
1053   \caption@teststar\caption@@freeze\@gobble\@firstofone}%

1054 \newcommand*\caption@@freeze[1]{%
1055   \global\let\caption@SCcontinued\relax
1056   \global\let\caption@SCsetup\@undefined
1057   \global\let\caption@SClentry\@undefined
1058   \global\let\caption@SCtext\@undefined
1059   \global\let\caption@SClabel\@undefined

1060   \let\caption@ORI@ContinuedFloat\ContinuedFloat
1061   \def\ContinuedFloat{%
1062     \caption@withoptargs\caption@SC@ContinuedFloat}%
1063   \def\caption@SC@ContinuedFloat##1{%
1064     \let\caption@ORI@setcounter\setcounter
1065     \let\caption@ORI@addtocounter\addtocounter
1066     \def\setcounter####1####2{\csname c@####1\endcsname####2\relax}%
1067     \def\addtocounter####1####2{\advance\csname c@####1\endcsname ####2\relax}%
1068     \caption@ORI@ContinuedFloat##1%
1069     \global\let\caption@SCcontinued\caption@ORI@ContinuedFloat
1070     \let\setcounter\caption@ORI@setcounter
1071     \let\addtocounter\caption@ORI@addtocounter}%
1072   \let\caption@ORI@setup\captionsetup
1073   \def\captionsetup{%
1074     \caption@withoptargs\caption@SC@setup}%
1075   \def\caption@SC@setup##1##2{%
1076     \caption@g@addto@list\caption@SCsetup{##2}%
1077     \caption@ORI@setup##1{##2}}%
1078   \let\caption@ORI\caption
1079   \def\caption{%
1080     \def\caption{\caption@Error{%
1081       Only one \noexpand\caption can be placed in this environment}}%
1082     \let\captionsetup\caption@setup
1083     \let\caption@@refstepcounter\caption@l@stepcounter
1084     \caption@ORI}%

```

```

1085 \long\def\@caption##1[##2]##3{%
1086 % \@bsphack
1087 \gdef\caption@SClentry{##2}%
1088 \gdef\caption@SCtext{##3}%
1089 % \@esphack}%
1090 \ignorespaces}%
1091 #1{% is \@gobble in star form, and \@firstofone otherwise
1092 \def\label##1{\@bsphack\gdef\caption@SClabel{##1}\@esphack}}%
1093 }%

\caption@defrost \caption@defrost
1094 \newcommand*\caption@defrost{%
1095 \ifx\caption@ORI@ContinuedFloat\@undefined
1096 \caption@defrost@setup
1097 \ifx\caption@SCtext\@undefined \else
1098 \expandafter\expandafter\expandafter\caption
1099 \expandafter\expandafter\expandafter[%
1100 \expandafter\expandafter\expandafter{%
1101 \expandafter\caption@SClentry\expandafter}\expandafter]%
1102 \expandafter{\caption@SCtext}%
1103 \fi
1104 \ifx\caption@SClabel\@undefined \else
1105 \expandafter\label\expandafter{\caption@SClabel}%
1106 \fi
1107 \else
1108 \caption@Error{Internal Error:\MessageBreak
1109 \noexpand\caption@defrost in same group as \string\caption@freeze}%
1110 \fi}%

1111 \newcommand*\caption@defrost@setup{%
1112 \caption@SCcontinued
1113 \ifx\caption@SCsetup\@undefined \else
1114 \expandafter\captionsetup\expandafter{\caption@SCsetup}%
1115 \fi}%

1116 }{}%
1117 \caption@undefbool{needfreeze}}

```

## 16.1 The float package

The float package usually do not use the  $\text{\LaTeX}$  kernel command `\@caption` to typeset the caption but `\float@caption` instead. (`\@caption` will only be used if the float is re-styled with `\restylefloat*`.)

The main two things `\float@caption` is doing different are:

- The caption will be typeset inside a `\savebox` called `\@floatcapt` so it can be placed above or below the float contents afterwards.
- `\@makecaption` will not be used to finally typeset the caption. Instead `\@fs@capt` will be used which definition is part of the float style. (Note that `\@fs@capt` will not typeset any vertical space above or below the caption; instead this space will be typeset by the float style code itself.)



```

1118 \caption@ifpackageloaded{float}[2001/11/08 v1.3d]{%
1119 \@ifpackageloaded{floatrow}{%
1120 \caption@ifpackageloaded{floatrow}[2007/08/24 v0.2a]{}{}%
1121 }{}%

```

`\@float@setevery` `\@float@setevery{<float type>}` is provided by the float package; it's called every time a floating environment defined with `\newfloat` or `\restylefloat` begins. We use this hook to do some adaptations and to setup the proper caption style (if defined) and additional settings declared with `\captionsetup[<float style>]`.

```

1122 \let\caption@ORI@float@setevery\@float@setevery
1123 \def\@float@setevery#1{%
1124 \float@ifcaption{#1}{%

```

First of all we set the caption position to it's proper value by converting `\@fs@iftopcapt` (which is part of a float style and controls where the caption will be typeset, above or below the float contents) to our `position=` setting. Since the spacing above and below the caption will be done by the float style and *not* by us this sounds quite useless. But in fact it isn't, since some packages based on the caption package (like the subfig package) could have an interest for this information and therefore use the `\caption@iftop` macro we provide in our kernel. Furthermore we need this information for ourself in `\captionof` which uses `\@makecaption` to finally typeset the caption with skips.

```

1125 \caption@setposition{\@fs@iftopcapt t\else b\fi}%

```

Afterward we redefine `\caption@setfloatcapt` (which will be used inside `\@caption`) so the caption will be set inside the box `\@floatcapt`, without extra vertical space.

```

1126 \renewcommand\caption@setfloatcapt[1]{%
1127 \let\@makecaption\caption@@make
1128 \global\setbox\@floatcapt\vbox{%
1129 \color@begingroup ##1\color@endgroup}}%

```

To allow different caption styles for different float styles we also determine the current float style (e.g. 'ruled') and select a caption style (and additional settings) with the same name, if defined.

```

1130 \float@getstyle\float@style{#1}%
1131 \caption@setstyle*\float@style
1132 \caption@setoptions\float@style
1133 }{}%
1134 \caption@freezeHref % will be defrosted in \float@makebox
1135 \caption@ORI@float@setevery{#1}}%

```

`\caption@typehook` L<sup>A</sup>T<sub>E</sub>X and almost every other packages use `\<type>`name to provide a macro for the type resp. environment name – for example the command `\figurename` will usually contain the name of the floating environment figure:

```

\newcommand\figurename{Figure}

```

But the float package doesn't follow this common naming convention: For floats defined with `\newfloat` it uses `\fname@<type>` instead, which breaks with our code (and with `\autoref` and some other things as well). So we have to map the float package name to the common one here.

*Note:* If the float was not defined with `\newfloat` but with `\restylefloat` instead, `\fname@<type>` is not defined.

```

1136 \g@addto@macro\caption@typehook{%

```

```

1137     \expandafter\ifx\csname #1name\endcsname\relax
1138     \expandafter\let\csname #1name\expandafter\endcsname
1139         \csname fname@#1\endcsname
1140     \fi}%

\fs@plaintop \fs@boxed Since the float styles plaintop and boxed don't use \abovecaptionskip
which could be set with skip= (plaintop uses \belowcaptionskip instead of
\abovecaptionskip, and boxed uses a fixed space of 2pt) we patch the according
float style macros here to change this.

1141 \g@addto@macro\fs@plaintop{\def\@fs@mid{\vspace\abovecaptionskip\relax}}%
1142 \g@addto@macro\fs@boxed{\def\@fs@mid{\kern\abovecaptionskip\relax}}%

\float@ifstyle \float@ifstyle{<type>}{<if-clause>}{<else-clause>}
Checks if the given <type> (e.g. figure) is associated with a float style (e.g. boxed).

1143 \providecommand*\float@ifstyle[1]{%
1144     \expandafter\ifx\csname fst@#1\endcsname\relax
1145     \expandafter\@secondoftwo
1146     \else
1147     \expandafter\@firstoftwo
1148     \fi}%

\float@getstyle \float@getstyle{<cmd>}{<type>}
Determining the float style is not so easy because the only hint provided by the float
package is the macro \fst@<float type> which points to the macro which represents the
float style. So for example after

\floatstyle{ruled}
\newfloat{Program}{tbp}{lop}

\fst@Program will be defined as

\def\fst@Program{\fs@ruled} .

So here is what we do: We make the first level expansion of \fst@<float type> a string
so we can gobble the first four tokens (= \fs@), so only the the name of the float style is
left.
TODO: We need to convert the catcodes here.

1149 \providecommand*\float@getstyle[2]{%
1150     \edef#1{%
1151         \noexpand\expandafter\noexpand\@gobblefour\noexpand\string
1152         \expandafter\expandafter\expandafter\noexpand
1153         \csname fst@#2\endcsname}%
1154     \edef#1{#1}%
1155     \caption@Debug{floatstyle{#2} = '#1'}}%

\float@setstyle \float@setstyle{<type>}{<style>}
Sets or changes the float style associated with <type>.

1156 \providecommand*\float@setstyle[2]{%
1157     \expandafter\edef\csname fst@#1\endcsname{%
1158         \expandafter\noexpand\csname fs@#2\endcsname}%

\float@dostyle \float@dostyle{<type>}

1159 \providecommand*\float@dostyle[1]{%
1160     \@nameuse{fst@#1}\@float@setevery{#1}}%

```

```

\float@ifcaption \float@ifcaption{<type>}{<if-clause>}{<else-clause>}
Here we determine if the user has used \newfloat resp. \restylefloat, or
\restylefloat*. This is quite easy: If \@float@c@<captype> is the same as
\float@caption, the user has used \newfloat or \restylefloat, otherwise
we assume he has used \restylefloat*. (This test will fail if some package re-
defines \float@caption, so we have to assume that there is no one.)

1161 \providecommand*\float@ifcaption[1]{%
1162   \expandafter\ifx\csname @float@c@#1\endcsname\float@caption
1163   \expandafter\@firstoftwo
1164   \else
1165   \expandafter\@secondoftwo
1166   \fi}%

1167 }{%
1168   \providecommand*\float@ifstyle[1]{\@secondoftwo}%
1169   \providecommand*\float@ifcaption[1]{\@secondoftwo}%
1170 % \clearcaptionsetup{boxed}% used by the floatrow package?
1171 }

```

The skip between ‘boxed’ floats and their caption defaults to 2pt.

```
1172 \captionsetup[boxed]{skip=2pt} % do not issue a warning when not used
```

To emulate the ‘ruled’ definition of \@fs@capt we provide a caption style ‘ruled’ with appropriate options. But if the package option ruled was specified, we setup some caption parameters to emulate the behavior of the caption package v1.x option ruled instead, i.e., the current caption settings will be used, but without margin and without ‘single-line-check’.

```

1173 \caption@ifbool{ruled}{%
1174   \captionsetup[ruled]{margin=0pt,minmargin=0,slc=0}%
1175 }{%
1176   \DeclareCaptionStyle{ruled}{labelfont=bf,labelsep=space,strut=0}%
1177 }
1178 \caption@undefbool{ruled}

```

## 16.2 The floatflt package

```
1179 \caption@ifPackageLoaded{floatflt}[1996/02/27 v1.3]{%
```

\floatingfigure We patch \floatingfigure so \caption@floatflt will be used.

```

1180 \let\caption@ORI@floatingfigure\floatingfigure
1181 \def\floatingfigure{%
1182   \caption@floatflt{figure}%
1183   \caption@ORI@floatingfigure}%

```

\floatingtable Same with \floatingtable...

```

1184 \let\caption@ORI@floatingtable\floatingtable
1185 \def\floatingtable{%
1186   \caption@floatflt{table}%
1187   \caption@setautoposition b%
1188   \caption@ORI@floatingtable}%

```

\caption@floatflt Here we do two things:

1. We use `\caption@setoptions{floating<type>}` so `\captionsetup[floating<type>]{...}` is supported.
2. `\linewidth` must be set correctly. Usually this is done by `\@parboxrestore` inside `\@caption`, but since we use `\@caption@boxrestore` we have to map this to `\@parboxrestore` instead.

```

1189 \newcommand*\caption@floatflt[1]{%
1190   \caption@settype{#1}%
1191   \caption@clearmargin
1192   \caption@setfullparboxrestore
1193   \caption@setoptions{floating#1}}%
1194 }{}

```

### 16.3 The fltpage package

```

1195 \caption@ifpackageloaded{fltpage}[1998/10/29 v.0.3]{%
1196   \caption@setbool{needfreeze}{1}%

```

`\FP@helpNote` Original code:

```

\newcommand{\FP@helpNote}[2]{%
  \typeout{FP#1 is inserted on page \pageref{#2}!}}%

```

```

1197 \renewcommand\FP@helpNote[2]{%
1198   \begingroup % save \caption@thepage
1199   \caption@pageref{#2}%
1200   \typeout{FP#1 is inserted on page \caption@thepage!}%
1201   \endgroup}%

```

`\FP@floatBegin` Original code:

```

\newcommand{\FP@floatBegin}[1]{%
  \gdef\@captive{#1}%
  \global\let\FP@savedCaptionCommand\caption%
  \global\let\FP@savedLabelCommand\label%
  \ifthenelse{\equal{\@captive}{figure}}{
    {\global\let\old@Fnum\Fnum@figure}%
    {\global\let\old@Fnum\Fnum@table}%
  }
  \let\FP@LabelText\@empty%
  \let\FP@CaptionText\@empty%
  \let\FP@optionalCaptionText\@empty%
  \renewcommand\label[1]{\gdef\FP@LabelText{##1}}%
  \renewcommand\caption[2][1]{%
    \gdef\FP@optionalCaptionText{##1}\gdef\FP@CaptionText{##2}}%
  \begin{lrbox}{\FP@floatCorpusBOX}%
} %

```

```

1202 \renewcommand*\FP@floatBegin[1]{%
1203   \def\@captive{#1}%
1204   \let\FP@LabelText\@empty
1205   \begin{lrbox}{\FP@floatCorpusBOX}%
1206   \caption@ifFPrefcap
1207     {\caption@freeze\relax}%

```

```

1208      {\def\label##1{\@bsphack\gdef\FP@LabelText{##1}\@esphack}%
1209      \caption@freeze*}%
1210      \ignorespaces}%

```

\FP@floatEnd **Original code:**

```

\newcommand{\FP@floatEnd}{%
  \end{lrbox}%
  \global\setbox\FP@floatCorpusBOX=\box\FP@floatCorpusBOX
  \stepcounter{FP@\@cctype C}%
  \FP@savedLabelCommand{\FP@positionLabel}%
  \FP@helpNote{\@cctype}{\FP@positionLabel}%
  \FP@float
    {\FP@positionLabel}% location label test
    {\begin{\@cctype}[p!]}
      \usebox{\FP@floatCorpusBOX}%
      \refstepcounter{\@cctype}%
      \ifthenelse{\equal{\FP@LabelText}{\@empty}}{
        {}{\FP@savedLabelCommand{\expandafter\protect\FP@LabelText}}}%
      \end{\@cctype}}
    {\addtocounter{\@cctype}{-1}}
    {\begin{\@cctype}[b!]}%
      \ifthenelse{\equal{\FP@guide}{\@empty}}{
        {}{\ifthenelse{\equal{\@cctype}{figure}}{
          {\renewcommand{\fnum@figure}{\old@Fnum\ \FP@guide}}}%
          {\renewcommand{\fnum@table}{\old@Fnum\ \FP@guide}}}%
        \setlength{\abovecaptionskip}{2pt plus2pt minus 1pt} % length above caption
        \setlength{\belowcaptionskip}{2pt plus2pt minus 1pt} % length above caption
        \FP@separatorCaption%
        \ifthenelse{\equal{\FP@optionalCaptionText}{\@empty}}{
          {\FP@savedCaptionCommand{\expandafter\protect\FP@CaptionText}}}%
          {\FP@savedCaptionCommand[\expandafter\protect\FP@optionalCaptionText]%
            {\expandafter\protect\FP@CaptionText}}}%
      \end{\@cctype}}%
  }%

1211 \renewcommand*\FP@floatEnd{%
1212   \end{lrbox}%
1213   \stepcounter{FP@\@cctype C}%
1214   \caption@label\FP@positionLabel
1215   \FP@helpNote\@cctype\FP@positionLabel
1216   \edef\FP@RestoreCounter{%
1217     \noexpand\setcounter{\@cctype}{\the\value\@cctype}%
1218     \noexpand\setcounter{ContinuedFloat}{\the\value{ContinuedFloat}}}%
1219   \FP@float
1220     {\FP@positionLabel}% location label test
1221     {\begin{\@cctype}[p!]}%
1222       \usebox\FP@floatCorpusBOX
1223       \caption@defrost@setup
1224       \caption@ifFPlistcap
1225         {\caption@refstepcounter\@cctype
1226           \expandafter\caption@makecurrent\expandafter\@cctype
1227             \expandafter{\caption@SClentry}}}%

```

```

1228         {\expandafter\captionlistentry\expandafter{\caption@SCLentry}}%
1229         \caption@makeanchor\relax
1230         \ifx\FP@LabelText\@empty \else
1231             \expandafter\label\expandafter{\FP@LabelText}%
1232         \fi
1233     \end\@capttype}%
1234     {\FP@RestoreCounter
1235     \@ifundefined{theH\@capttype}{}{%
1236         \expandafter\l@addto@macro\csname theH\@capttype\endcsname{.FP}}}%
1237     {\begin\@capttype[b!]%
1238         \let\FP@savedSetfnumCommand\caption@setfnum
1239         \def\caption@setfnum##1{%
1240             \FP@savedSetfnumCommand{##1}%
1241             \ifx\FP@guide\@empty \else
1242                 \expandafter\l@addto@macro\csname fnum@##1\endcsname{\ {\FP@guide}}%
1243             \fi}%
1244         \setlength\abovecaptionskip{2pt plus 2pt minus 1pt}% length above caption
1245         \setlength\belowcaptionskip{2pt plus 2pt minus 1pt}% length below caption
1246         \caption@setoptions{FP\@capttype}%
1247         \FP@separatorCaption
1248         \caption@ifFPlistcap{}{\let\caption@addcontentsline\@gobbletwo}%
1249         \caption@defrost
1250     \end\@capttype}%
1251 }%

1252 \caption@For{typelist}{%
1253     \newcounter{FP@#1C}%
1254     \newenvironment{FP#1}{\FP@floatBegin{#1}}{\FP@floatEnd}}%
1255 }%

1256 \let\caption@ifFPlistcap\@undefined
1257 \let\caption@ifFPrefcap\@undefined
1258 }

```

## 16.4 The hyperref package

```

1259 \caption@IfPackageLoaded{hyperref}[2003/11/30 v6.74m]{%
1260     % Test if hyperref has stopped early
1261     \@ifundefined{ifhyp@stoppedearly}{%
1262         \newif\ifhyp@stoppedearly
1263         \@ifundefined{H@refstepcounter}\hyp@stoppedearlytrue{%
1264             \@ifundefined{hyper@makecurrent}\hyp@stoppedearlytrue{%
1265                 \@ifundefined{measuring@true}\hyp@stoppedearlytrue{}}}{%
1266             \ifhyp@stoppedearly % hyperref has stopped early
1267                 \caption@InfoNoLine{%
1268                     Hyperref support is turned off\MessageBreak
1269                     because hyperref has stopped early}%
1270             \else
1271                 %
1272                 \g@addto@macro\caption@prepareslc{\measuring@true}%
1273             \fi

```

`\caption@@refstepcounter` We redefine `\caption@@refstepcounter` so `\H@refstepcounter` will be used instead of `\refstepcounter` inside `\caption` & `\captionlistentry`.

```

1273         \renewcommand*\caption@@refstepcounter{\H@refstepcounter}%

```

`\caption@makecurrent` We redefine `\caption@makecurrent` so a `hyperref` label will be defined inside `\@caption`.  
*Note:* Will be redefined by `\caption@start`.

```

1274 \renewcommand*\caption@makecurrent[2]{%
1275 \caption@makecurrentHref{#1}%
1276 \caption@Debug{hyperref current=\@currentHref}%
1277 \caption@getttitle{#2}}%
1278 \newcommand*\caption@makecurrentHref{\hyper@makecurrent}%

```

`\caption@makeanchor` We redefine `\caption@makeanchor` so a `hyperref` anchor will be set inside `\@caption`.  
*Note:* Will be redefined by `\caption@start`.

```

1279 \renewcommand\caption@makeanchor[1]{%
1280 \caption@Debug{hyperref anchor: \@currentHref}%
1281 % If we cannot have nesting, the anchor is empty.
1282 \ifHy@nesting
1283 \expandafter\hyper@@anchor\expandafter{\@currentHref}{#1}%
1284 \else
1285 \Hy@raisedlink{%
1286 \expandafter\hyper@@anchor\expandafter{\@currentHref}{\relax}%
1287 }#1%
1288 \fi}%
1289 \g@addto@macro\caption@prepareslc{\let\caption@makeanchor\@firstofone}%

```

### The `hypcap` option

`\if@capstart` Like the `hypcap` package we define the switch `\if@capstart`, too.

```

1290 \newif\if@capstart

```

`\caption@start` While the `hypcap` package defines a macro called `\capstart` our variant is called `\caption@start` and is controlled by the option `hypcap=false/true`.

```

1291 \def\caption@start{\caption@ifhypcap\caption@start@relax}%
1292 \def\caption@start@{%

```

Generate the `hyperref` label and set the `hyperref` anchor, usually (if `hypcap=false`) both is done inside `\@caption`.

```

1293 \caption@makestart\@captype
1294 \caption@startanchor\@currentHref

```

Prevent `\@caption` from generating a new `hyperref` label, use the label we save in `\hc@currentHref` instead. (We also support the `@capstart` flag from the `hypcap` package.)

```

1295 \global\@capstarttrue
1296 \let\hc@currentHref\@currentHref
1297 \def\caption@makecurrentHref##1{%
1298 \global\@capstartfalse
1299 \global\let\@currentHref\hc@currentHref}%

```

Prevent `\@caption` from generating a `hyperref` anchor since this has already been done.

```

1300 \let\caption@makeanchor\@firstofone
1301 }%

```

`\caption@makestart` `\caption@makestart{<type>}` defines a `hyperref` anchor inside `\caption@start`. Since we offer `\ContinuedFloat` the float counter can change between ‘now’ and `\caption`, i.e., we simply don’t know the figure or table counter yet and therefore we are not able to generate the ‘right’ `hyperref` label. Two different solutions of this problem came into my mind:

1. I could use the aux file for this purpose.
- or-
2. I set `hypertextnames=false` locally. Furthermore I use `#1.caption.<counter>` (instead of `#1.<counter>`) as naming scheme for `\@currentHref` to avoid conflicts with other hyper links which are generated with `hypertextnames=true`.

The first idea has the advantage that the ‘right’ anchor name will be generated, but one needs an additional `LATEX` run if figures or tables will be inserted or removed.

The second idea has the advantage that it’s very easy to implement, but has some side-effects, e.g. the anchor names don’t follow the figure or table label names anymore.

Since I’m lazy I implemented the second idea, maybe I will revise this later on.

```

1302   \newcommand*\caption@makestart[1]{%
1303     \begingroup
1304       \Hy@hypertextnamesfalse
1305   %   \gdef\@currentHlabel{}%
1306       \hyper@makecurrent{#1.caption}%
1307     \endgroup
1308     \caption@Debug{hypcap start=\@currentHref}}%
```

`\caption@startanchor` `\caption@startanchor{<Href>}` sets a `hyperref` anchor inside `\caption@start`. This code was taken from the `hypcap` package[10] and adapted.

*Note:* Since `\hyper@@anchor{<Href>}{\relax}` can cause a change from vertical mode to horizontal mode (design flaw in `hyperref` package!?), and since the workaround `\let\leavevmode\relax` which can be found in the `hypcap` package is not always sufficient (for example with “Direct pdfmark support” and `breaklinks=true`), we use `\caption@anchor` instead of `\hyper@@anchor` here.

```

1309   \newcommand*\caption@startanchor[1]{%
1310     \ifvmode\begingroup
1311       \caption@Debug{hypcap anchor: #1 (vertical mode)}%
1312       \@tempdima\prevdepth
1313       \nointerlineskip
1314       \vspace*{-\caption@hypcapspace}%
1315       \caption@anchor{#1}%
1316       \vspace*{\caption@hypcapspace}%
1317       \prevdepth\@tempdima
1318     \endgroup\else
1319       \caption@Debug{hypcap anchor: #1 (horizontal mode)}%
1320       \caption@anchor{#1}%
1321     \fi}%%
```

`\caption@anchor` `\caption@anchor{<Href>}` sets a `hyperref` anchor.

```

1322   \newcommand*\caption@anchor[1]{%
1323     \ifmeasuring@ \else
1324       \caption@raisedlink{\hyper@anchorstart{#1}\hyper@anchorend}%
1325     \fi}%%
```



*Note:* Since `\Hy@raisedlink` change `\@tempdima` we surrounded it by `\ifvmode`, suppressing “LaTeX Warning: Float too large for page by 1.0pt” in sideways floats. (This is not necessary since `hyperref v6.77`.)

```
1326 \ifx\HyperRaiseLinkLength\@tempdima
1327 \def\caption@raisedlink#1{\ifvmode#1\else\Hy@raisedlink{#1}\fi}%
1328 \else
1329 \let\caption@raisedlink\Hy@raisedlink
1330 \fi
```

`\caption@@start` Will be used by `\caption@freezeHref`. Apart from that we issue a warning if we expect a saved `hyperref` label coming from `\caption@start`, but there isn’t any.

```
1331 \def\caption@@start{%
1332 \ifundefined{hc@currentHref}{%
1333 \caption@Warning{%
1334 The option ‘hycap=true’ will be ignored for this\MessageBreak
1335 particular \string\caption}}}%%
```

`\caption@freezeHref` Suppress `\caption@start` from generating a `hyperref` label and setting a `hyperref` anchor. Instead if `\@caption` generates a `hyperref` label, it will be stored in `\caption@currentHref`. Furthermore we need to redefine `\caption@setfloatcapt` so no `hyperref` anchor will be placed in `\@caption`.

```
1336 \def\caption@freezeHref{%
1337 \let\caption@ORI@start\caption@start
1338 \def\caption@start{\let\caption@start\caption@ORI@start}%
1339 %
1340 % \l@addto@macro\caption@subtyphook{%
1341 % \let\caption@@start\caption@ORI@@start}%
1342 %
1343 \global\let\caption@currentHref\@undefined
1344 \def\caption@@start{\global\let\caption@currentHref\@currentHref}%
1345 %
1346 \let\caption@ORI@setfloatcapt\caption@setfloatcapt
1347 \renewcommand*\caption@setfloatcapt{%
1348 \ifx\caption@currentHref\@undefined \else
1349 \let\caption@makeanchor\@firstofone
1350 \fi
1351 \caption@ORI@setfloatcapt}}%
```

`\caption@defrostHref` If there is a frozen `\@currentHref`, we set the `hyperref` anchor here.

```
1350 \def\caption@defrostHref{%
1351 \ifx\caption@currentHref\@undefined \else
1352 \caption@startanchor\caption@currentHref
1353 \global\let\caption@currentHref\@undefined
1354 \fi}%%
```

`\float@makebox` Do our own redefinition of `\float@makebox`, if it was redefined by the `hyperref` package.

```
1355 \ifundefined{HyOrg@float@makebox}{}{%
1356 \caption@Debug{%
1357 Redefining \noexpand\float@makebox (again)\@gobble}%
1358 \let\caption@ORI@float@makebox\float@makebox % save for compatibility mode
1359 \renewcommand\float@makebox[1]{%
1360 \HyOrg@float@makebox{#1}\relax \caption@defrostHref}}%
1361 }%
```

```
1362 \fi}{}
```

## 16.5 The hypcap package

```
1363 \caption@ifpackageloaded{hypcap}{% v1.0
1364 \ifx\caption@start\relax \else % hyperref hasn't stopped early
```

If the hypcap package was loaded, we give up our own hyperlink placement algorithm and give the control over the placement to the hypcap package instead.

`\capstart` We do this simply by mapping `\capstart` to `\caption@start@`, although our code does not behave exactly like the original one: The original `\capstart` has an effect on the next `\caption` only but our version affects *all* `\captions` in the same environment, at least unless a new `\capstart` will be placed.

```
1365 \let\caption@ORI@capstart\capstart % save for compatibility mode
1366 \@ifundefined{capstarttrue}% check for v1.10 of hypcap package
1367 {\def\capstart{\caption@start@}}%
1368 {\def\capstart{\ifcapstart\caption@start@\fi}}%
1369 \let\caption@start\relax
1370 \let\caption@@start\relax
```

`\caption@hypcapspace` Furthermore we map our `\caption@hypcapspace` to `\hypcapspace` offered by the hypcap package.

```
1371 \caption@set@bool\caption@ifhypcap 1%
1372 \renewcommand*\caption@hypcapspace{\hypcapspace}%
1373 \fi}{}
```

## 16.6 The listings package

```
1374 \caption@ifpackageloaded{listings}[2004/02/13 v1.2]{%
```

`\lst@MakeCaption` To support the listings package we need to redefine `\lst@MakeCaption` so the original stuff is nested with `\caption@begin` and `\caption@end` etc.

*Note:* This macro is always called twice (with 't' resp. 'b' as parameter), therefore we need an extra group here.

```
1375 \let\caption@ORI@lst@MakeCaption\lst@MakeCaption
1376 \def\lst@MakeCaption#1{% #1 is 't' or 'b'
1377 \begingroup
```

Workaround for bug in listings package: If `\hsize` seems not to be set correctly, we set it to `\linewidth`.

```
1378 \ifdim\hsize>\linewidth
1379 \hsize\linewidth
1380 \fi
```

First of all, we set `position=#1` and if it was set to 'top', we swap the skips so the default behavior of the listings package will not be changed. (Note that the listings package has set its own `\abovecaptionskip` & `\belowcaptionskip` values prior to calling `\lst@MakeCaption`.)

```
1381 \caption@setposition{#1}%
1382 \caption@iftop{%
1383 \@tempdima\belowcaptionskip
1384 \belowcaptionskip\abovecaptionskip
1385 \abovecaptionskip\@tempdima}{}%
```

Workaround for issue with wrong skips (should be examined further)

```
1386 \caption@setup{rule=0}%
```

Afterwards we set the local ‘lstlisting’ options.

```
1387 \caption@setoptions{lstlisting}%
```

If the position= is now set to auto, we take over the captionpos= setting from the listings package.

```
1388 \caption@setautoposition{#1}%
```

At the end we do similar stuff as in our \@caption code.

```
1389 \caption@begin{lstlisting}%
```

```
1390 \caption@ORI@lst@MakeCaption{#1}%
```

```
1391 \caption@end
```

```
1392 \endgroup}%
```

\lst@makecaption Wrapper macros for typesetting the caption= resp. title= value.

```
\lst@maketitle 1393 \def\lst@makecaption{\caption@starfalse\@makecaption}%
```

```
1394 \def\lst@maketitle{\caption@startrue\@makecaption\@empty}%
```

\ext@lstlisting Since the listings package do not define \ext@lstlisting, but we needed it when \captionof{lstlisting} will be done by the end user, we define it here.

```
1395 \providecommand*\ext@lstlisting{lol}%
```

```
1396 {} }
```

## 16.7 The longtable package

\LTcapttype \LTcapttype is preset to table.

```
1397 \providecommand*\LTcapttype{table}
```

```
1398 \caption@IfPackageLoaded{longtable}[1995/05/24 v3.14]{%
```

```
1399 \RequirePackage{ltcaption}[2007/09/01]%
```

```
1400 \let\LT@makecaption\@undefined
```

\LT@array We redefine \LT@array here to get \captionsetup{<options>} working inside longtables.

*Note:* Since the hyperref package patches \LT@array as well and since this only works with the original definition of \LT@array, we have to do this after the hyperref package, i.e. \AtBeginDocument.

```
1401 \caption@AtBeginDocument{%
```

```
1402 \let\caption@ORI@LT@array\LT@array
```

```
1403 \renewcommand*\LT@array{%
```

\captionsetup for longtable:

```
1404 \global\let\caption@opt@@longtable\@undefined
```

```
1405 \def\captionsetup{%
```

```
1406 \noalign\bgroup
```

```
1407 \@ifstar\@captionsetup\@captionsetup}% gobble *
```

```
1408 \def\@captionsetup##1{\LT@captionsetup{##1}\egroup}%
```

```
1409 \def\LT@captionsetup##1{%
```

```
1410 \captionsetup@startrue\caption@setup@options[@longtable]{##1}%
```

```
1411 \global\let\caption@opt@@longtable\caption@opt@@longtable}%
```

```

\captionabove & \captionbelow for longtable: (KOMA-Script document class)
1412 \def\@captionabovetrue{\LT@captionsetup{position=t}}%
1413 \def\@captionabovefalse{\LT@captionsetup{position=b}}%
\captionlistentry for longtable:
1414 \def\captionlistentry{%
1415 \noalign\bgroup
1416 \@ifstar{\egroup\LT@captionlistentry}% gobble *
1417 {\egroup\LT@captionlistentry}}%
1418 \def\LT@captionlistentry##1{%
1419 \caption@listentry\@firstoftwo[\LTcapttype]{##1}}%
\ContinuedFloat for longtable:
(Commented out, since it's not deeply tested and quite useless anyway)
Note: hyperref versions < v6.76j uses 2x \hyper@makecurrent
1420 % \caption@ifhyccap{%
1421 % \let\caption@ORI@hyper@makecurrent\hyper@makecurrent
1422 % \def\hyper@makecurrent##1{%
1423 % \let\hyper@makecurrent\caption@ORI@hyper@makecurrent
1424 % \caption@makestart{##1}%
1425 % \let\Hy@LT@currentHlabel\@currentHlabel
1426 % \let\Hy@LT@currentHref\@currentHref
1427 % \def\hyper@makecurrent###1{%
1428 % \let\@currentHlabel\Hy@LT@currentHlabel
1429 % \let\@currentHref\Hy@LT@currentHref}}%
1430 % \let\caption@ORI@ContinuedFloat\ContinuedFloat
1431 % \def\ContinuedFloat{\noalign{%
1432 % \gdef\caption@setContinuedFloat{%
1433 % \let\caption@resetContinuedFloat\@gobble}%
1434 % \def\caption@setoptions####1{%
1435 % \g@addto@macro\caption@setContinuedFloat{%
1436 % \caption@setoptions{####1}}}%
1437 % \let\@capttype\LTcapttype
1438 % \caption@ORI@ContinuedFloat}}%
1439 % }{%
1440 % \def\ContinuedFloat{\noalign{%
1441 % \caption@Error{%
1442 % \noexpand\ContinuedFloat inside longtables\MessageBreak
1443 % is only available with 'hyccap=true'}}}%
1444 % }%
1445 % \global\let\caption@setContinuedFloat\@empty
1446 % \def\ContinuedFloat{\noalign{%
1447 % \caption@Error{\noexpand\ContinuedFloat outside float}}}%
1448 % \caption@ORI@LT@array}}%

```

\LT@c@ption The original implementation:

```

\def\LT@c@ption#1[#2]#3{%
\LT@makecaption#1\fnun@table{#3}%
\def\@tempa{#2}%
\ifx\@tempa\@empty\else
{\let\\\space
\addcontentsline{lot}{table}{\protect\numberline{\thetable}{#2}}}%
\fi}

```

Our implementation uses `\LTcapttype` instead of `{table}`:

```
1449 \long\def\LT@caption#1[#2]#3{%
1450     \LT@makecaption#1{\csname fnum@\LTcapttype\endcsname}{#3}%
1451     \LT@captionlistentry{#2}}%
```

```
\LT@makecaption \LT@makecaption{\langle cmd\rangle}{\langle label\rangle}{\langle text\rangle}
```

The original definition:

```
\def\LT@makecaption#1#2#3{%
  \LT@mcol\LT@cols c{\hbox to\z@{\hss\parbox[t]\LTcapwidth{%
    % Based on article class "\@makecaption", "#1" is "\@gobble" in star
    % form, and "\@firstofone" otherwise.
    \sbox\@tempboxa{#1{#2: }#3}%
    \ifdim\wd\@tempboxa>\hsize
      #1{#2: }#3%
    \else
      \hbox to\hsize{\hfil\box\@tempboxa\hfil}%
    \fi
    \endgraf\vskip\baselineskip}%
  \hss}}}
```

Our definition:

```
1452 \renewcommand\LT@makecaption[3]{%
1453     \caption@LT@make{%
```

If `\LTcapwidth` is not set to its default value 4in we assume that it shall overwrite our own setting. (But `\captionsetup{longtable}{width=...}` will overwrite `\LTcapwidth`.)

```
1454     \caption@settype*\LTcapttype
1455     \ifdim\LTcapwidth=4in \else
1456         \setcaptionwidth\LTcapwidth
1457     \fi
1458     \caption@setoptions{longtable}%
1459 %     \caption@setContinuedFloat
1460     \caption@setoptions{@longtable}%
```

`position=auto` is a bad idea for longtables, but we do our very best. This works quite well for captions inside the longtable contents, but not for captions inside the longtable (end)foot.

*Note:* This should be ‘top’ if unclear!

```
1461     \caption@setautoposition{\ifcase\LT@rows t\else b\fi}%
```

We set `\ifcaption@star` according the 1st argument.

```
1462     \caption@startrue#1\caption@starfalse
1463     \caption@resetContinuedFloat\LTcapttype
1464     \caption@begin\LTcapttype
1465     \caption@normalsize
```

The following skip has the purpose to correct the height of the `\parbox[t]`. Usually it's the height of the very first line, but because of our extra skips (`\abovecaptionskip` and `\belowcaptionskip`) it's always 0pt.

(A different idea would be typesetting the first skip outside the longtable column with `\noalign{\vskip...}`, but this means we have to move `\caption@begin` to some

other place because it does not work in tabular mode. And at the moment I have no idea on how to do this in an elegant way...)

```
1466 \vskip-\ht\strutbox
```

The following code should look familiar. We do our skips and use `\caption@@make` to typeset the caption itself.

```
1467 \caption@iftop{\vskip\belowcaptionskip}{\vskip\abovecaptionskip}%
1468 \caption@@make{#2}{#3}\endgraf
1469 \caption@iftop{\vskip\abovecaptionskip}{\vskip\belowcaptionskip}%
1470 \caption@end}%
1471 {} }
```

## 16.8 The picinpar package

```
1472 \caption@ifPackageLoaded{picinpar}{%
```

`\figwindow` The **picinpar** package comes with its own caption code (`\wincaption`, `@wincaption`, `\tabwindow` `@makewincaption`, ...) so we redefine `\figwindow` & `\tabwindow` to use `\caption` instead.

```
1473 \long\def\figwindow[#1,#2,#3,#4] {%
1474 \caption@window{figure}%
1475 \caption@setoptions{figwindow}%
1476 \begin{window}[#1,#2,{#3},\caption@wincaption{#4}] }%
1477 \long\def\tabwindow[#1,#2,#3,#4] {%
1478 \caption@window{table}%
1479 \caption@setoptions{tabwindow}%
1480 \begin{window}[#1,#2,{#3},\caption@wincaption{#4}] }%
```

`\caption@window` Beside calling `\caption@settype` we redefine `\caption@boxrestore` (as in `floatflt` & `picins` package support) and `@makecaption` (as in `float` package support) here.

```
1481 \newcommand*\caption@window[1]{%
1482 \let\@makecaption\caption@@make
1483 \caption@setautoposition b%
1484 \caption@settype{#1}%
1485 \caption@clearmargin
1486 \caption@setfullparboxrestore}%

```

`\caption@wincaption` This one finally typesets the caption using `\caption`.

```
1487 \newcommand\caption@wincaption[1]{%
```

This will be done twice for every `figwindow` & `tabwindow` caption – on the first run `\picwd` is 0pt, on the second run `\picwd` is `\hsize`.

```
1488 \ifdim\picwd=\z@
1489 \let\caption@makecurrent\@gobbletwo
1490 \let\caption@@start\relax
1491 \caption@prepareslc
1492 \fi
```

The argument `#1` could contain simply the caption text (e.g. A figure caption), but it could also contain an optional argument, the *lst.entry* (e.g. [An entry to the L<sup>OF</sup>] {A figure caption}). Therefore we have to test if `#1` begins with `[` or not; furthermore we support a starred variant – as in `\caption*` – so we test for `*`, too.

```

1493 \edef\@tempa{\expandafter\noexpand\@car#1\@nil}%
1494 \if\@tempa*%
1495     \let\@tempa\@firstofone
1496 \else\if\@tempa[%
1497     \let\@tempa\@firstofone
1498 \else
1499     \let\@tempa\@empty
1500 \fi\fi
1501 \expandafter\caption\@tempa{#1}}%

1502 {}{}

```

## 16.9 The picins package

`\piccaptiontype` `\piccaptiontype{<type>}`

We offer this macro for changing the `<type>` of the caption, so the user doesn't have to redefine `\@caption`, as proposed in the `picins` documentation.

*Note:* We define this macro here so it can be used in the preamble of the document, even when the caption package was loaded prior to the `picins` package.

```
1503 \newcommand*\piccaptiontype[1]{\def\@piccaptiontype{#1}}
```

```
1504 \caption@ifpackageloaded{picins}{%
```

Initial set `\@piccaptiontype` and undefine `\@captiontype` which was set to figure by the `picins` package.

```

1505 \@ifundefined{@piccaptiontype}{%
1506     \caption@iftype{%
1507         \let\@piccaptiontype\@captiontype
1508     }%
1509     \def\@piccaptiontype{figure}%
1510 }%
1511 {}{}%
1512 \let\@captiontype\@undefined

```

`\piccaption` The original code:

```
\def\piccaption{\@ifnextchar [{\@piccaption}{\@piccaption[]}}
```

Our code uses `\caption@star` so `\piccaption*` works, and `\caption@dblarg` so `\piccaption{}` works correctly.

```
1513 \def\piccaption{\caption@star\relax\caption@dblarg\@piccaption}}%
```

`\make@piccaption` The original code:

```

\def\make@piccaption{%
    [...]
    \setbox\@TEXT=\vbox{\hsize\hsiz@\caption[\sh@rtf@rm]{\capti@nt@xt}}%
}

```

In our code we have to correct several things:

1. `\@captiontype` must be defined, since we have removed the global definition.

2. We use `\caption@setoptions{parpic}` so `\captionsetup[parpic]{...}` is supported.
3. `\linewidth` must be set correctly. Usually this is done by `\@parboxrestore` inside `\@caption`, but since we use `\@caption@boxrestore` we have to map this to `\@parboxrestore` instead.
4. The two arguments of `\caption (\sh@rtf@rm & \capti@nt@xt)` should be expanded on first level so `\caption[] {...}` and `\caption[...]{}` work correctly.

```

1514 \let\caption@ORI@make@piccaption\make@piccaption
1515 \def\make@piccaption{%
1516   \let\caption@ORI\caption
1517   \long\def\caption[##1]##2{%
1518     \caption@freezeHref % will be defrosted in \ivparpic
1519     \caption@settype\@piccaptiontype
1520 %   \ifnum\c@piccaptionpos>2\relax
1521     \caption@clearmargin
1522 %   \else
1523 %     \captionwidth\z@ % do not use "width=" setting
1524 %   \fi
1525     \caption@setfullparboxrestore
1526     \caption@setoptions{parpic}%
1527     \caption@setautoposition b%
1528     \expandafter\expandafter\expandafter\caption@ORI
1529       \expandafter\expandafter\expandafter[%
1530         \expandafter\expandafter\expandafter{%
1531           \expandafter##1\expandafter}\expandafter]\expandafter{##2}}%
-or- \begingroup
      \toks0\expandafter{##1} \toks2\expandafter{##2}
      \edef\x{\endgroup
        \noexpand\caption@ORI[{\the\toks0}]{\the\toks2}}
      \x
-or- \edef\x{%
      \noexpand\caption@ORI[{\unexpanded\expandafter{##1}}]%
        {\unexpanded\expandafter{##2}}}
      \x
1532 \caption@ORI@make@piccaption
1533 \let\caption\caption@ORI}%

```

`\ivparpic` We need to set our hyperref anchor here. Not bullet-proof since we have to redefine `\noindent` here!

```

1534 \let\caption@ORI@ivparpic\ivparpic
1535 \def\ivparpic(#1,#2)(#3,#4)[#5][#6]#7{%
1536   \let\caption@ORI@noindent\noindent
1537   \def\noindent{%
1538     \caption@defrostHref
1539     \let\noindent\caption@ORI@noindent
1540     \noindent}%
1541   \caption@ORI@ivparpic(#1,#2)(#3,#4)[#5][#6]{#7}%
1542   \let\noindent\caption@ORI@noindent}%

```



```

1543 }{%
1544 \let\piccaptiontype\@undefined
1545 }

```

## 16.10 The rotating package

```

1546 \caption@ifPackageLoaded{rotating}[1995/08/22 v2.10]{%
\rotcaption Make \rotcaption* work.
1547 \def\rotcaption{\let\@makecaption\@makerotcaption\caption}%
1548 % \let\@rotcaption\@undefined

\rotcaptionof Make \rotcaptionof(*) work.
1549 \def\rotcaptionof{%
1550 \caption@teststar\caption@of{\rotcaption*}\rotcaption}%

\@makerotcaption Original (bugfixed) code:

\long\def\@makerotcaption#1#2{%
\setbox\@tempboxa\hbox{#1: #2}%
\ifdim \wd\@tempboxa > .8\vsiz
\rotatebox{90}{%
\begin{minipage}{.8\textheight}#1: #2\end{minipage}%
}%\par % <== \par removed (AR)
\else%
\rotatebox{90}{\box\@tempboxa}%
\fi
\nobreak\hspace{12pt}% <== \nobreak added (AR)
}

```

Our version emulates this behavior, but if width= is set, the rotated caption is always typeset as minipage. (Note that margin= is not supported here.)

```

1551 \long\def\@makerotcaption#1#2{%
1552 \ifdim\captionwidth=\z@
1553 \setcaptionwidth{.8\textheight}%
1554 \caption@slc{#1}{#2}{.8\vsiz}{%
1555 \let\caption@makerot\caption@@make
1556 \caption@clearmargin
1557 % \long\def\caption@parbox##1##2{\hbox{\hsiz=.8\textheight\relax##2}}%
1558 % (not needed because \rotatebox uses an \hbox anyway)
1559 \let\caption@parbox\@secondoftwo}%
1560 \caption@set@bool\caption@ifslc0% been there, done that
1561 \fi

1562 \rotatebox{90}{\caption@makerot{#1}{#2}}%
1563 \nobreak\hspace{12pt}}%

1564 \newcommand\caption@makerot[2]{%
1565 \begin{minipage}\captionwidth\caption@@make{#1}{#2}\end{minipage}}%

1566 \caption@For{typelist}{%
1567 \newenvironment{sideways#1}{\@rotfloat{#1}}{\end@rotfloat}%
1568 \newenvironment{sideways#1*}{\@rotdblfloat{#1}}{\end@rotdblfloat}}%

1569 }{}

```

## 16.11 The sidecap package

```

1570 \caption@ifpackageloaded{sidecap}[1999/05/11 v1.4d]{%
1571   \caption@setbool{needfreeze}{1}%

\SC@caption First of all, we let sidecap use a current definition of \caption.
              (This is only required for version 1.5d of the sidecap package.)
1572   \caption@atbegindocument{\let\SC@caption=\caption}%

\SC@zfloat This macro will be called at the start of the environment, here is a good opportunity to do
            some adaptations to \caption and \captionsetup.
1573   \let\caption@ORI@SC@zfloat\SC@zfloat
1574   \def\SC@zfloat#1#2#3[#4]{%

            First we use the original definition, but save & restore \caption so \caption@freeze
            will work correctly.
1575       \let\caption@ORI\caption
1576       \caption@ORI@SC@zfloat{#1}{#2}{#3}[#4]%
1577       \let\caption\caption@ORI

            Since the sidecap package uses our \caption code outside the environment the reg-
            ular \captionsetup will not work. So we need a special version here which saves
            the given argument list which will be executed later on. Furthermore we need to make
            \caption* work.
1578       \caption@settype*{#2}%
1579       \caption@freeze*}%

\endSC@FLOAT This macro will be called at the end of the environment, here we need to setup our stuff
              before the sidecap package actually typesets its caption.
1580   \let\caption@ORI@endSC@FLOAT\endSC@FLOAT
1581   \def\endSC@FLOAT{%

            Note: \@captype isn't defined here, this will be done inside the original definition of
            \endSC@FLOAT. But \SC@captype is defined and can be used here, if needed.

1582       \let\caption@ORI@settype\caption@settype
1583       \def\caption@settype##1{% will be done in \xfloat
1584         \caption@ORI@settype*{##1}% do not change \@currentlabel
1585         \caption@setSC@justify
1586         %%% \caption@setoptions{SCfloat}%
1587         \caption@setoptions{SC\@captype}%
1588         \caption@start}%

            Before we can typeset the caption we need to set the margin to zero because any extra
            margin would only be disturbing here.
            (We don't need to take care about the caption position because the sidecap package set
            both \abovecaptionskip and \belowcaptionskip to a skip of zero anyway.)
            Furthermore \SC@justify will override the caption justification, if set. The usage of
            \SC@justify differs from version to version of the sidecap package:
            Version 1.4: \SC@justify is not defined
            Version 1.5: \SC@justify is \relax when not set
            Version 1.6: \SC@justify is \@empty when not set

1589       \def\caption@setSC@justify{%
1590         \caption@clearmargin
1591         \@ifundefined{SC@justify}{}{}%
```

```

1592         \ifx\SC@justify\@empty \else
1593             \let\caption@hj\SC@justify
1594             \let\SC@justify\@empty
1595         \fi}}%

```

Make the original definition of `\endSC@FLOAT` to use our caption stuff instead of its own.

*Note:* At this point the `sidecap` definition of `\caption` is valid, not the regular one!

```

1596     \let\caption\SC@orig@caption
1597     \def\SC@orig@caption[##1]##2{\caption@defrost}%

```

Finally we call the original definition of `\endSC@FLOAT`.

```

1598     \caption@setSC@justify % for compatibility mode
1599     \caption@ORI@endSC@FLOAT}%

1600 \newcommand*\caption@For@SC[2]{%
1601     \def#1{b}% = \sidecaptionvpos{#2}{b} (v1.6)
1602     \newenvironment{SC#2}%
1603         {\SC@float[#1]{#2}}{\endSC@float}%
1604     \newenvironment{SC#2*}%
1605         {\SC@dblfloat[#1]{#2}}{\endSC@dblfloat}}%
1606 \@onlypreamble\caption@For@SC
1607 \caption@For{typelist}{%
1608     \expandafter\caption@For@SC\csname SC@#1@vpos\endcsname{#1}}%
1609 }{}

```

## 16.12 The subfigure package

```

1610 \caption@ifPackageLoaded{subfigure}[2002/01/23 v2.1]{%

```

`\sf@ifpositiontop` If the `subfigure` package is loaded, we map `\sf@ifpositiontop` to `\iffiguretopcap` resp. `\iftabletopcap`, so the `subfigure v2.1` options `figbotcap` etc. will still work.

```

1611 \def\sf@ifpositiontop{%
1612     \ifx\@captype\@undefined
1613         \expandafter\@gobbletwo
1614     \else\ifx\@captype\relax
1615         \expandafter\expandafter\expandafter\@gobbletwo
1616     \else
1617         \expandafter\expandafter\expandafter\sf@if@position@top
1618     \fi\fi}

1619 \def\sf@if@position@top{%
1620     \@ifundefined{if\@captype topcap}%
1621         {\@gobbletwo}%
1622         {\@nameuse{if\@captype topcap}}%
1623     \expandafter\@firstoftwo
1624 \else
1625     \expandafter\@secondoftwo
1626 \fi}}

1627 }{}

```

## 16.13 The supertabular and xtab packages

```

1628 \caption@ifpackageloaded{supertabular}[2002/07/19 v4.1e]{%
\tablecaption Make \topcaption* and \bottomcaption* work.
1629   \renewcommand*\tablecaption{%
1630     \caption@star
1631     {\refstepcounter{table}}%
1632     {\caption@dblarg{\@xtablecaption}}}%

\@xtablecaption Make \nameref and \autoref work.
1633   \let\caption@ORI@xtablecaption\@xtablecaption
1634   \long\def\@xtablecaption[#1]#2{%
1635     \caption@gettitle{#2}%
1636     \caption@ORI@xtablecaption[#1]{#2}}%

\ST@caption The original code:

\long\def\ST@caption#1[#2]#3{\par%
  \addcontentsline{\csname ext@#1\endcsname}{#1}%
    {\protect\numberline{
      \csname the#1\endcsname}{\ignorespaces #2}}
  \begingroup
    \@parboxrestore
    \normalsize
    \if@topcaption \vskip -10\p@ \fi
    \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
    \if@topcaption \vskip 10\p@ \fi
  \endgroup

1637   \long\def\ST@caption#1[#2]#3{\par%
1638     \caption@settype*{#1}%
1639     \caption@setoptions{supertabular}%

The position= setting will be overwritten by the supertabular package: If \topcaption
was used, the position will be top automatically, bottom otherwise.

1640   \def\caption@fixposition{%
1641     \caption@setposition{\if@topcaption t\else b\fi}}%

1642   \caption@beginex{#1}{#2}{#3}%
1643   \caption@boxrestore
1644   \caption@normalsize
1645   \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
1646   \caption@end}%

1647 }{}

1648 \caption@ifpackageloaded{xtab}[2000/04/09 v2.3]{%

\tablecaption Make \topcaption* and \bottomcaption* work.
1649   \renewcommand*\tablecaption{%
1650     \caption@star
1651     {\refstepcounter{table}}%
1652     {\caption@dblarg{\@xtablecaption}}}%

```

```

\@xtablecaption Make \nameref and \autoref work.
1653 \let\caption@ORI@xtablecaption\@xtablecaption
1654 \long\def\@xtablecaption[#1]#2{%
1655     \caption@getttitle{#2}%
1656     \caption@ORI@xtablecaption[#1]{#2}}%

\ST@caption The original code:

\long\def\ST@caption#1[#2]#3{\par%
    \@initisotab
    \addcontentsline{\csname ext@#1\endcsname}{#1}%
        {\protect\numberline{#3}
            \csname the#1\endcsname}{\ignorespaces #2}}%

    \begingroup
        \@parboxrestore
        \normalsize
    %% \if@topcaption \vskip -10\p@ \fi
        \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
    %% \if@topcaption \vskip 10\p@ \fi
    \endgroup
    \global\advance\ST@pageleft -\PWSTcapht
    \ST@trace\tw@{Added caption. Space left for xtabular: \the\ST@pageleft}}%

1657 \long\def\ST@caption#1[#2]#3{\par%
1658     \caption@settype*{#1}%
1659     \caption@setoptions{xtabular}%

1660     \def\caption@fixposition{%
1661         \caption@setposition{\if@topcaption t\else b\fi}}%

1662     \@initisotab
1663     \caption@beginex{#1}{#2}{#3}%
1664     \caption@boxrestore
1665     \caption@normalsize
1666     \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
1667     \caption@end
1668     \global\advance\ST@pageleft -\PWSTcapht
1669     \ST@trace\tw@{Added caption. Space left for xtabular: \the\ST@pageleft}}%

1670 {} }

```

## 16.14 The threeparttable package

```

1671 \caption@IfPackageLoaded{threeparttable}[2003/06/13 v3.0]{%

\threeparttable Unfortunately \@capytype is not set when \TPT@common will be used, so we have to
redefine \threeparttable and \measuredfigure instead.

1672 \let\caption@ORI@threeparttable\threeparttable
1673 \renewcommand*\threeparttable{%
1674     \caption@settype{table}%
1675     \caption@setposition a% ?
1676     \caption@clearmargin
1677     \caption@setoptions{threeparttable}%
1678     \caption@ORI@threeparttable}%

```

\measuredfigure Same here...

```
1679 \let\caption@ORI@measuredfigure\measuredfigure
1680 \renewcommand*\measuredfigure{%
1681   \caption@settype{figure}%
1682   \caption@setposition a% ?
1683   \caption@clearmargin
1684   \caption@setoptions{measuredfigure}%
1685   \caption@ORI@measuredfigure}%
```

\TPT@caption The original code:

```
\def\TPT@caption#1[#2]#3{\gdef\TPT@docapt
{\par\global\let\TPT@docapt\@undefined \TPT@LA@caption{#1}[{#2}]}%
{\strut\ignorespaces#3\ifhmode\unskip\@finalstrut\strutbox\fi}}%
\ifx\TPT@hsize\@empty \let\label\TPT@gatherlabel \abovecaptionskip\z@skip
\else \TPT@docapt \fi \ignorespaces}

1686 \def\TPT@caption#1[#2]#3{%
1687   \gdef\TPT@docapt{%
1688     \global\let\TPT@docapt\@undefined
1689     \caption@setautoposition\caption@TPT@position
1690     \TPT@LA@caption{#1}[{#2}]{#3}}%
1691   \ifx\TPT@hsize\@empty
1692     \let\label\TPT@gatherlabel % Bug: does not work for measuredfigures
1693     \gdef\caption@TPT@position{t}%
1694     \g@addto@macro\TPT@docapt\caption@TPT@eatvskip
1695   \else
1696     \def\caption@TPT@position{b}%
1697     \TPT@docapt
1698   \fi
1699   \ignorespaces}%
1700 %\newcommand*\caption@TPT@eatvskip{\vskip-.2\baselineskip}%
1701 \def\caption@TPT@eatvskip#1\vskip{#1\@tempdima=}%
1702 {} }
```

## 16.15 The wrapfig package

```
1703 \caption@ifPackageLoaded{wrapfig}{% ver 3.3 (Oct 12, 1999)
```

\float@ifstyle \float@ifstyle{<type>}{<if-clause>}{<else-clause>}  
(see float package support for details)

```
1704 \providecommand*\float@ifstyle[1]{%
1705   \expandafter\ifx\csname fst@#1\endcsname\relax
1706     \expandafter\@secondoftwo
1707   \else
1708     \expandafter\@firstoftwo
1709   \fi}%
```

\caption@restylewrapfloat This one redefines the wrap#1 environment, e.g. wrapfigure. Our code uses  
\caption@setoptions{wrapfigure} so \captionsetup[wrapfigure]{...}  
will work.

But first we check if our redefinition was already done, this could happen inside `\float@restyle` when the `wrapfig` support of the `float` package was not installed successfully, so it has not redefined `\wrap#1` there.

```

1710 \newcommand*\caption@restylewrapfloat[1]{%
1711   \expandafter\ifx\csname caption@OUR@wrap#1\expandafter\endcsname
1712     \csname wrap#1\endcsname
1713   \caption@Error{%
1714     For a successful cooperation of the 'wrapfig' package\MessageBreak
1715     with the 'float' package you should load the 'wrapfig'\MessageBreak
1716     package *after* the 'float' package}%
1717   \else
1718     \expandafter\let\csname caption@ORI@wrap#1\expandafter\endcsname
1719     \csname wrap#1\endcsname
1720     \@namedef{wrap#1}{\caption@wrapfloat{#1}}%
1721     \expandafter\let\csname caption@OUR@wrap#1\expandafter\endcsname
1722     \csname wrap#1\endcsname
1723   \fi}%

```

`\caption@wrapfloat`

```

1724 \newcommand*\caption@wrapfloat[1]{%
1725   \caption@settype*{#1}%
1726   \float@ifstyle{#1}{%
1727     \ifx\WF@floatstyhook\@undefined
1728       \caption@Error{%
1729         For a successful cooperation of the 'wrapfig' package\MessageBreak
1730         with the 'float' package you should use at least\MessageBreak
1731         'wrapfig' version 3.6}%
1732     \else
1733       \float@dostyle{#1}%
1734       \fi}}%
1735   \caption@clearmargin
1736   \caption@setoptions{wrapfloat}%
1737   \caption@setoptions{wrap#1}%
1738   \@nameuse{caption@ORI@wrap#1}}%

```

Now we redefine the `wrapfig` environments we know about.

If someone has placed a `\newfloat` right between `\usepackage{wrapfig}` and `\usepackage{caption}` (or loads the `caption` package first, so all these patches will be done with `\AtBeginDocument`) we have bad luck since the `float` package do not offer a list of (re)styled floats. (This would finally lead to an error in `\caption@setfloatcapt`.)

```

1739 \caption@restylewrapfloat{figure}%
1740 \caption@restylewrapfloat{table}%
1741 \caption@For{typelist}{%
1742   \newenvironment{wrap#1}{\wrapfloat{#1}}{\endwrapfloat}%
1743   \caption@restylewrapfloat{#1}}%
1744 \ifx\WF@floatstyhook\@undefined \else % wrapfig v3.6

```

`\float@restyle` If the `wrapfig` package v3.6 is used, we patch `\float@restyle` (if defined), too, so new or restyled floats will be handled correctly, too.

```

1745   \@ifundefined{float@restyle}{}{%
1746     \toks@=\expandafter{\float@restyle{#1}% (env may or may not be defined)

```

```

1747         \caption@restylewrapfloat{#1}}%
1748         \edef\@tempa{\def\noexpand\float@restyle##1{\the\toks@}}%
1749         \@tempa}% perform redefinitions

\wrapfloat An additional check of the package load order: If both, neither the wrapfig package
nor the caption package haven't catch \float@restyle, we finally splash down at
\wrapfloat.
1750     \let\caption@ORI@wrapfloat\wrapfloat
1751     \def\wrapfloat#1{%
1752         \float@ifstyle{#1}{%
1753             \caption@Error{%
1754                 For a successful cooperation of the 'wrapfig' package\MessageBreak
1755                 with the 'float' package you should load the 'wrapfig'\MessageBreak
1756                 package *right after* the 'float' package}}}%
1757         \caption@ORI@wrapfloat{#1}}%

1758     \fi                                     % wrapfig v3.6

\WF@rapt We place our hyperref anchor here.
Original code:
\def\WF@rapt[#1]#2{% final two args: #1 = overhang, #2 = width,
\gdef\WF@ovh{#1}% hold overhang for later, when \width is known
\global\setbox\WF@box\top\bggroup \setlength\hsize{#2}%
\ifdim\hsize>\z@ \@parboxrestore \else
\setbox\z@\hbox\bggroup \let\wf@@caption\caption \let\caption\wf@caption
\ignorespaces \fi}

Our code:
1759 \def\WF@rapt[#1]#2{% final two args: #1 = overhang, #2 = width,
1760     \gdef\WF@ovh{#1}% hold overhang for later, when \width is known
1761     \global\setbox\WF@box\top\bggroup \setlength\hsize{#2}%
1762     \caption@start
1763     \ifdim\hsize>\z@ \@parboxrestore \else
1764     \setbox\z@\hbox\bggroup \let\wf@@caption\caption \let\caption\wf@caption
1765     \ignorespaces \fi}%

1766 } {}

```



## References

- [1] Frank Mittelbach and Michel Goossens:  
*The L<sup>A</sup>T<sub>E</sub>X Companion (2nd. Ed.)*,  
Addison-Wesley, 2004.
- [2] Till Tantau:  
*User Guide to the Beamer Class, Version 3.07*,  
March 11, 2007
- [3] Markus Kohm & Jens-Uwe-Morawski:  
*KOMA-Script – a versatile L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub> bundle*,  
2007-01-09
- [4] Victor Eijkhout:  
*An introduction to the Dutch L<sup>A</sup>T<sub>E</sub>X document classes*,  
3 September 1989
- [5] Anselm Lingnau:  
*An Improved Environment for Floats*,  
2001/11/08
- [6] Mats Dahlgren:  
*Welcome to the floatflt package*,  
1998/06/05
- [7] Olga Lapko:  
*The floatrow package documentation*,  
2007/08/24
- [8] Sebastian Gross:  
*Welcome to the beta test of fltpage package!*,  
1998/11/13
- [9] Sebastian Rahtz & Heiko Oberdiek:  
*Hypertext marks in L<sup>A</sup>T<sub>E</sub>X*,  
November 12, 2007
- [10] Heiko Oberdiek:  
*The hypcap package – Adjusting anchors of captions*,  
2007/04/09
- [11] Carsten Heinz & Brooks Moses:  
*The Listings Package*,  
2007/02/22
- [12] David Carlisle:  
*The longtable package*,  
2004/02/01
- [13] Friedhelm Sowa:  
*Pictures in Paragraphs*,  
July 13, 1993

- [14] Joachim Bleser and Edmund Lang:  
*PicIns-Benutzerhandbuch Version 3.0*,  
September 1992
- [15] Sebastian Rahtz and Leonor Barroca:  
*A style option for rotated objects in  $\text{\LaTeX}$* ,  
1997/09/26
- [16] Rolf Niepraschk & Hubert Gäßlein:  
*The sidecap package*,  
2003/06/06
- [17] Steven D. Cochran:  
*The subfigure package*,  
2002/07/02
- [18] Steven D. Cochran:  
*The subfig package*,  
2005/07/05
- [19] Johannes Braams and Theo Jurriens:  
*The supertabular environment*,  
2002/07/19
- [20] Donald Arseneau:  
*Three part tables: title, tabular environment, notes*,  
2003/06/13
- [21] Donald Arseneau:  
*WRAPFIG.STY ver 3.6*,  
2003/01/31
- [22] Peter Wilson:  
*The xtab package*,  
2004/05/24