

The Implementation of the caption package^{*}

Axel Sommerfeldt

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

| | |
|---------------------------------|--|
| caption-eng.pdf | The English documentation |
| caption-rus.pdf | The Russian documentation ¹ |
| caption-deu.pdf | The German documentation |

^{*}This package has version number v3.2b, last revised 2011/08/18.

¹Thanks a lot to Olga Lapko for this translation

Contents

| | |
|--|-----------|
| 1 Identification | 4 |
| 2 Loading the kernel | 4 |
| 3 Check against incompatible document classes | 4 |
| 4 Check against incompatible packages | 4 |
| 5 Declaration of options | 4 |
| 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 |
| 6 $\mathcal{A}\mathcal{M}\mathcal{S}$ & SMF document classes support | 7 |
| 7 KOMA-Script document classes support | 7 |
| 8 Processing of options | 9 |
| 9 <code>\caption</code>, <code>\@caption</code>, and <code>\@makecaption</code> | 9 |
| 10 <code>\captionof</code> and <code>\captionlistentry</code> | 18 |
| 11 <code>\captionbox</code> | 21 |
| 12 <code>\ContinuedFloat</code> | 22 |
| 13 Internal helpers | 23 |
| 14 Support for sub-captions | 26 |
| 15 Document class & Babel package support | 28 |
| 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 |

| | |
|--|-----------|
| 16 Package support | 30 |
| 16.1 The float package | 33 |
| 16.2 The floatflt package | 36 |
| 16.3 The fltpage package | 36 |
| 16.4 The hyperref package | 39 |
| 16.5 The hypcap package | 42 |
| 16.6 The listings package | 43 |
| 16.7 The longtable package | 44 |
| 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 | 52 |
| 16.13 The supertabular and xtab packages | 52 |
| 16.14 The threeparttable package | 54 |
| 16.15 The wrapfig package | 55 |

1 Identification

```
1 \NeedsTeXFormat{LaTeX2e}[1994/12/01]
2 \ProvidesPackage{caption}[2011/08/18 v3.2b 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   \begingroup
43     \caption@ifundefined\language\language\language{}{}%
44     \caption@ifundefined\caption\language\language{}{}%
45     \expandafter\g@addto@macro\csname captions\language\language\endscsname
46       {\caption@setname{#1}{#2}}}%
47   \endgroup}
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   \DeclareCaptionOptionNoValue{#1without}{\caption@Within{#1}{none}}%

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

55 \DeclareCaptionOption{within}{%
56   \caption@ifundefined\c@figure{}{\caption@Within{figure}{#1}}%
57   \caption@ifundefined\c@table{}{\caption@Within{table}{#1}}%
58   \caption@For{typelist}{\caption@Within{#1}{#1}}%
59   \def\caption@within@default{#1}}
60 \DeclareCaptionOptionNoValue{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 `hycap=off` one can turn the `hycap` support off (default is on).

```

134 \DeclareCaptionOption{hycap}[1]{\caption@setbool{hycap}{#1}}
135 \DeclareCaptionOption{hycapSPACE}{\def\caption@hycapSPACE{#1}}
136 \captionsetup{hycap=1,hycapSPACE=.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 \caption@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     \caption@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}}%
227     \caption@withoptargs\@gobbletwo}}

```

`\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.

```

228 \newcommand*\caption@star[2]{%
229   \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^AT_EX PR latex/2472](#).)

```

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

```

`\caption@prepareanchor`

```

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

```

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

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

```

252 \long\def\caption@makecaption#1#2{%
253   \caption@iftop
254     {\vskip\belowcaptionskip}%
255     {\caption@rule\vskip\abovecaptionskip}%
256   \caption@@make{#1}{#2}%
257   \caption@iftop
258     {\vskip\abovecaptionskip\caption@rule}%
259     {\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.

```

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

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

263   \caption@CheckCommand\caption{%
264     % ltfloat.dtx [2002/10/01 v1.1v LaTeX Kernel (Floats)]
265     \def\caption{%
266       \ifx\@capttype\@undefined
267         \@latex@error{\noexpand\caption outside float}\@ehd
268         \expandafter\@gobble
269       \else
270         \refstepcounter\@capttype
271         \expandafter\@firstofone
272       \fi
273       {\@dblarg{\@caption\@capttype}}%
274     }%

275   \caption@CheckCommand\caption{%
276     % beamerbaselocalstructure.sty,v 1.53 2007/01/28 20:48:21 tantau
277     \def\caption{
278       \ifx\@capttype\@undefined
279         \@latex@error{\noexpand\caption outside figure or table}\@ehd
280         \expandafter\@gobble
281       \else
282         \refstepcounter\@capttype
283         \expandafter\@firstofone
284       \fi
285       {\@dblarg{\@caption\@capttype}}%
286     }%

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

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

```

```

306 \def\caption{%
307   \ifx\@capttype\@undefined
308     \latex@error{\noexpand\caption outside float}\@ehd
309     \expandafter\@gobble
310   \else
311     \H@refstepcounter\@capttype
312     \@ifundefined{fst@\@capttype}{%
313       \let\Hy@tempa\@caption
314     }{%
315       \let\Hy@tempa\Hy@float@caption
316     }%
317     \expandafter\@firstofone
318   \fi
319   {\@dblarg{\Hy@tempa\@capttype}}%
320 }%

321 \caption@CheckCommand\caption{%
322   % hyperref.sty [2007/08/05 v6.76j Hypertext links for LaTeX]
323   \def\caption{%
324     \ifx\@capttype\@undefined
325       \latex@error{\noexpand\caption outside float}\@ehd
326       \expandafter\@gobble
327     \else
328       \H@refstepcounter\@capttype
329       \let\Hy@tempa\@caption
330       \@ifundefined{float@caption}{%
331         }{%
332           \expandafter\ifx\csname @float@c@\@capttype\endcsname\float@caption
333             \let\Hy@tempa\Hy@float@caption
334           \fi
335         }%
336       \expandafter\@firstofone
337     \fi
338     {\@dblarg{\Hy@tempa\@capttype}}%
339   }%

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

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

```

```

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

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

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

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

```

```

412         \else
413         \Hy@raisedlink{\hyper@@anchor{\@currentHref}{\relax}}#3%
414         \fi
415     }%
416     \par
417 \endgroup
418 } }%

419 \caption@CheckCommand\@caption{%
420 % hyperref.sty [2007/04/09 v6.76a Hypertext links for LaTeX]
421 % hyperref.sty [2007/06/12 v6.76h Hypertext links for LaTeX]
422 % hyperref.sty [2007/08/05 v6.76j Hypertext links for LaTeX]
423 \long\def\@caption#1[#2]#3{%
424     \expandafter\ifx\csname if@capstart\expandafter\endcsname
425         \csname iftrue\endcsname
426     \global\let\@currentHref\hc@currentHref
427 \else
428     \hyper@makecurrent{\@cuptype}%
429 \fi
430 \def\@currentlabelname{#2}%
431 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
432     \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
433 }%
434 \begingroup
435     \@parboxrestore
436     \if@minipage
437         \@setminipage
438     \fi
439     \normalsize
440     \expandafter\ifx\csname if@capstart\expandafter\endcsname
441         \csname iftrue\endcsname
442     \global\@capstartfalse
443     \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
444 \else
445     \@makecaption{\csname fnum@#1\endcsname}{%
446         \ignorespaces
447         \ifHy@nesting
448             \hyper@@anchor{\@currentHref}{#3}%
449         \else
450             \Hy@raisedlink{\hyper@@anchor{\@currentHref}{\relax}}#3%
451         \fi
452     }%
453 \fi
454 \par
455 \endgroup
456 } }%

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

```

```

465 \fi
466 \def\@currentlabelname{#2}%
467 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
468 \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
469 }%
470 \begingroup
471 \par\boxrestore
472 \if@minipage
473 \setminipage
474 \fi
475 \normalsize
476 \expandafter\ifx\csname if@capstart\expandafter\endcsname
477 \csname iftrue\endcsname
478 \global\@capstartfalse
479 \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
480 \else
481 \@makecaption{\csname fnum@#1\endcsname}{%
482 \ignorespaces
483 \ifHy@nesting
484 \expandafter\hyper@anchor\expandafter{\@currentHref}{#3}%
485 \else
486 \Hy@raisedlink{%
487 \expandafter\hyper@anchor\expandafter{\@currentHref}{\relax}%
488 }%
489 #3%
490 \fi
491 }%
492 \fi
493 \par
494 \endgroup
495 }%

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

```

```

519         \normalsize
520         \expandafter\ifx\csname if@capstart\expandafter\endcsname
521             \csname iftrue\endcsname
522             \global\@capstartfalse
523             \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
524         \else
525             \@makecaption{\csname fnum@#1\endcsname}{%
526                 \ignorespaces
527                 \ifHy@nesting
528                     \expandafter\hyper@@anchor\expandafter{\@currentHref}{#3}%
529                 \else
530                     \Hy@raisedlink{%
531                         \expandafter\hyper@@anchor\expandafter{%
532                             \@currentHref
533                             }{\relax}%
534                     }%
535                     #3%
536                 \fi
537             }%
538         \fi
539     \par
540 \endgroup
541 }}%

542 \caption@CheckCommand\@caption{%
543     % nameref.sty [2006/12/27 v2.28 Cross-referencing by name of section]
544     \long\def\@caption#1[#2]{%
545         \def\@currentlabelname{#2}%
546         \NR@@caption{#1}[\@currentlabelname]{#2}%
547     }%

548 \caption@CheckCommand\@caption{%
549     % nameref.sty [2009/11/27 v2.32 Cross-referencing by name of section]
550     \long\def\@caption#1[#2]{%
551         \NR@getttitle{#2}%
552         \NR@@caption{#1}[\@currentlabelname]{#2}%
553     }%

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

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

```



```

572 %         \caption@setfloattype
573 %         \@capttype
574 %         \sf@ifpositiontop{%
575 %             \@listsubcaptions{#1}%
576 %             \sf@old@caption{#1}[\{#2\}]{#3}%
577 %         }{%
578 %             \sf@old@caption{#1}[\{#2\}]{#3}%
579 %             \@listsubcaptions{#1}%
580 %         }}%
581     }%

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

```

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

```

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

```

`\ContinuedFloat` is not supported in compatibility mode.

```

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

```

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

```

599 \caption@atbegindocument*{%
600     \let\caption@start\relax
601     \caption@ifundefined\caption@ORI@capstart{}{%
602         \caption@debug{%
603             Restore hypcap definition of \string\capstart\@gobble}%
604         \let\capstart\caption@ORI@capstart}%
605     \caption@ifundefined\caption@ORI@float@makebox{}{%
606         \caption@debug{%
607             Restore hyperref redefinition of \string\float@makebox\@gobble}%
608         \let\float@makebox\caption@ORI@float@makebox}%
609     }%

```

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

```

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

```

```

616         was given}%
617     }{}%

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

622 }%

623 \caption@AtBeginDocument*{%
624     \let\caption@ORI@capstart\@undefined
625     \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.

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

630 }

```

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.

```

631 \caption@AtBeginDocument{\caption@redefine}

\@makecaption
632 \let\@makecaption\caption@makecaption

```

10 `\captionof` and `\captionlistentry`

```

633 \caption@AtBeginDocument{%
634     \DeclareCaptionOption{type}{\setcaptiontype{#1}}%
635     \DeclareCaptionOption{type*}{\setcaptiontype*{#1}}%
636     \DeclareCaptionOptionNoValue{subtype}{\setcaptionsubtype\relax}%
637     \DeclareCaptionOptionNoValue{subtype*}{\setcaptionsubtype*}%
638 }

```

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.

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

```

`\setcaptionsubtype` Same, but sets the sub-type.

```

642 \newcommand\setcaptionsubtype{%
643   \caption@iftype
644   \caption@setsubtype
645   {\caption@Error{\noexpand\setcaptionsubtype outside float}}}%

646 \newcommand\caption@setsubtype{%
647   \@ifstar
648   {\caption@@settype{sub}*{sub\@caption}}%
649   {\caption@@settype{sub}{sub\@caption}}}%

```

`\caption@settype` `\caption@settype*{<type>}`

sets `\@caption` 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).

```

650 \newcommand*\caption@settype{%
651   \caption@@settype{}}

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

654 \newcommand*\caption@@@settype[3]{%
655   % #1 = "" or "sub"
656   % #2 = \@firstoftwo in star form, \@secondoftwo otherwise
657   % #3 = <type>, e.g. "figure" or "table"
658   \caption@Debug{#1type=#3}%
659   \caption@checkgrouplevel{#1}{%
660     \captionsetup{#1type#2*\@empty=...}#2{ or
661       \@backslashchar#1captionof}{}}%

662   \edef\caption@tempa{#3}%
663   \expandafter\ifx\csname @#1caption\endcsname\caption@tempa \else
664     \ifcaptionsetup@star\else\@nameuse{caption@#1type@warning}\fi
665     \fi
666   \expandafter\let\csname @#1caption\endcsname\caption@tempa
667   \@nameuse{caption@#1typehook}%

668   \caption@setoptions{#3}%
669   \ifx\caption@opt\relax
670     \@nameundef{caption@#1type@warning}%
671   \else
672     \@namedef{caption@#1type@warning}{\caption@Warning{%
673       The #1caption type was already set to
674       '\csname @#1caption\endcsname'\MessageBreak}}%
675     \fi

676   \let\caption@ifrefstepcounter\@secondoftwo
677   #2{}%
678   \let\@currentlabel\caption@undefinedlabel
679   % \let\@currentHlabel\@undefined
680   \ifx\caption@x@label\@undefined
681     \let\caption@x@label\label

```

```

682     \let\label\caption@xlabel
683     \fi
684     \caption@start}}

```

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

```

685 \newcommand*\caption@typehook{}

```

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

```

686 \newcommand*\caption@iftype{%
687   \caption@ifundefined\@capytype\@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 \mathcal{E} -TeX)

```

688 \begingroup\expandafter\expandafter\expandafter\endgroup
689 \expandafter\ifx\csname currentgrouplevel\endcsname\relax
690   \caption@Debug{TeX engine: TeX}
691   \let\caption@checkgrouplevel\@gobbletwo
692 \else
693   \caption@Debug{TeX engine: e-TeX}
694   \newcommand*\caption@checkgrouplevel[2]{%
695     \ifundefined{#1caption@grouplevel}{%
696       \caption@ifundefined\caption@grouplevel{\let\caption@grouplevel\z@}{}%
697       \ifnum\currentgrouplevel>\caption@grouplevel\relax
698         \expandafter\edef\csname #1caption@grouplevel\endcsname{%
699           \the\currentgrouplevel}%
700       \else
701         \caption@Warning{\string#2\MessageBreak outside box or environment}%
702       \fi
703     }{} }
704 \fi

```

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

```

705 \newcommand*\caption@undefinedlabel{%
706   \protect\caption@xref{\caption@labelname}{\on@line}}
707 \DeclareRobustCommand*\caption@xref[2]{%
708   \caption@WarningNoLine{\noexpand\label before \string\caption#2}%
709   \@setref\relax\@undefined{#1}}
710 \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.

```

711 \newcommand*\caption@xlabel[1]{%
712   \caption@@xlabel
713   \def\caption@labelname{#1}%
714   \caption@x@xlabel{#1}}
715 \newcommand*\caption@@xlabel{%
716   \global\let\caption@@xlabel\@empty
717   \@bsphack
718   \protected@write\@auxout{}%

```

```

719      {\string\providecommand*\string\caption@xref[2]{%
720        \string\@setref\string\relax\string\@undefined{\string##1}}}%
721      \@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)

722 \caption@AtBeginDocument{%
723   \def\captionof{\caption@teststar\caption@of{\caption*}\caption}}
724 \newcommand*\caption@of[2]{\setcaptiontype*{#2}#1}

\captionlistentry \captionlistentry[<float type>]{<list entry>}
\captionlistentry* [<float type>]{<list entry>}
725 \newcommand*\captionlistentry{%
726   \caption@teststar\@captionlistentry\@firstoftwo\@secondoftwo}

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

729 \def\caption@listentry#1[#2]#3{%
730   \@bsphack
731   #1{\caption@getttitle{#3}}%
732   {\caption@refstepcounter{#2}}%
733   \caption@makecurrent{#2}{#3}}%
734   \caption@addcontentsline{#2}{#3}%
735   \@esphack}

```

11 \captionbox

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

```

736 \newcommand*\captionbox{%
737   \let\captionbox@settype\@gobble
738   \caption@withoptargs\caption@box}

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

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

743 \long\def\caption@iibox#1#2#3[#4]#5{%
744   \setbox\@tempboxa\hbox{#5}%
745   \begingroup
746   \captionbox@settype*% set \caption@position
747   \caption@iftop{%
748     \endgroup
749     \parbox[t]{#3}{%
750       \captionbox@settype\relax
751       \caption@setposition t%
752       \vbox{\caption#1{#2}}%
753       \captionbox@hrule
754       \csname caption@hj@#4\endcsname
755       \unhbox\@tempboxa}%

```

```

756 }{%
757   \endgroup
758   \parbox[b]{#3}{%
759     \captionbox@settype\relax
760     \caption@setposition b%
761     \csname caption@hj@#4\endcsname
762     \unhbox\@tempboxa
763     \captionbox@hrule
764     \vtop{\caption#1{#2}}}%
765   }}

766 \newcommand*\captionbox@hj@default{c}
767 \newcommand*\captionbox@hrule{\hrule\@height\z@\relax}

768 \providecommand*\caption@hj@c{\centering}
769 \providecommand*\caption@hj@l{\raggedright}
770 \providecommand*\caption@hj@r{\raggedleft}
771 \providecommand*\caption@hj@s{}

```

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

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

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

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

```

772 \def\ContinuedFloat{%
773   \@ifnextchar[\@Continued@Float\@ContinuedFloat}

774 \def\@Continued@Float[#1]{\addtocounter{#1}{\m@ne}

775 \def\@ContinuedFloat{%
776   \caption@iftype
777     {\addtocounter{\@capttype}{\m@ne}
778     \caption@ContinuedFloat{\@capttype}%
779     {\caption@Error{\noexpand\ContinuedFloat outside float}}}}

```

```

780 \def\caption@ContinuedFloat#1{%
781   \@ifstar{\caption@Continued@Float@{#1}}{\caption@Continued@Float{#1}}
782 \def\caption@Continued@Float@{%
783   \addtocounter{\@cptype}\@ne
784   \@stpelt{ContinuedFloat}\stepcounter{ContinuedFloat}%
785   \def\caption@resetContinuedFloat##1{\xdef\caption@CFtype{##1}}%
786   \caption@@ContinuedFloat}

787 \def\caption@Continued@Float#1{%
788   \edef\caption@tempa{#1}%
789   \ifx\caption@tempa\caption@CFtype
790     \stepcounter{ContinuedFloat}%
791     \let\caption@resetContinuedFloat\@gobble
792     \caption@@ContinuedFloat{#1}%
793     \sf@ContinuedFloat{#1}%
794   \else
795     \caption@Error{Continued `#1' after `\caption@CFtype'}%
796   \fi}

797 \def\caption@@ContinuedFloat#1{%
798   \expandafter\l@addto@macro\csname the#1\endcsname\theContinuedFloat
799   \@ifundefined{theH#1}{}{%
800     \expandafter\l@addto@macro\csname theH#1\endcsname{%
801       \@alph@c@ContinuedFloat}}%
802   \caption@setoptions{ContinuedFloat}%
803   \caption@setoptions{continued#1}}

804 \providecommand*\sf@ContinuedFloat[1]{}
805 \newcommand*\caption@CFtype{??}

```

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

```

806 \newcounter{ContinuedFloat}
807 \let\theContinuedFloat\@empty

```

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

```

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

```

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

```

810 \newcommand\phantomcaption{%
811   \caption@iftype
812     {\caption@refstepcounter\@cptype}%
813     {\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.

```

814 \newcommand*\caption@refstepcounter[1]{%

```

```

815 \@ifundefined{c@#1}%
816   {\caption@Error{No float type ‘#1’ defined}}%
817   {\caption@resetContinuedFloat{#1}%
818    \caption@@refstepcounter{#1}%
819    \let\caption@ifrefstepcounter\@firstoftwo}}
820 \newcommand*\caption@@refstepcounter{\refstepcounter}
821 \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[]{}).
822 \caption@ifundefined\kernel@ifnextchar
823   {\newcommand\caption@dblarg[1]{\@ifnextchar[{#1}]{\caption@xdblarg{#1}}}%
824   {\newcommand\caption@dblarg[1]{\kernel@ifnextchar[{#1}]{\caption@xdblarg{#1}}}%
825   \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)

826 \newcommand*\caption@begin[1]{%
827   \begingroup
828   \caption@setfnum{#1}%
829   \caption@fixposition
830   \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>.

831 \newcommand\caption@beginex[3]{%
832   \caption@begin{#1}%
833   \let\lst@@caption\relax
834   \caption@addcontentsline{#1}{#2}%
835   \caption@ifempty{#3}{}

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

836 \newcommand*\caption@end{%
837   \endgroup
838   \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.

```



```

839 \newcommand*\caption@setfnum[1]{%
840   \@ifundefined{fnum#1}{\iftrue}{\ifx\caption@lfmt\caption@lfmt@default\else}%
841   \@namedef{fnum#1}{\caption@fnum{#1}}%
842   \fi}

\caption@setparboxrestore \caption@setparboxrestore{<partial or full}}
843 \newcommand*\caption@setparboxrestore[1]{%
844   \caption@ifinlist{#1}{full}{%
845     \let\caption@parboxrestore\@firstoftwo
846   }\caption@ifinlist{#1}{default,light,partial}{%
847     \let\caption@parboxrestore\@secondoftwo
848   }}%
849   \caption@Error{Undefined parboxrestore `#1'}%
850   }}}

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

853 \DeclareCaptionOption{parboxrestore}{\caption@setparboxrestore{#1}}
854 \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.
855 \newcommand*\caption@boxrestore{%
856   \caption@parboxrestore{\@parboxrestore}{%
857     \let\if@nobreak\iffalse
858     \let\if@noskipsec\iffalse
859     \let\par\@par
860 %   \let\-\@dischyph
861 %   \let'\@acci\let\''\@accii\let\=\@acciii
862     \parindent\z@ \parskip\z@skip
863     \everypar{}%
864 %   \linewidth\hsize
865 %   \@totalleftmargin\z@
866     \leftskip\z@skip \rightskip\z@skip \@rightskip\z@skip

```

```

867 \parfillskip\@flushglue \lineskip\normallineskip
868 \baselineskip\normalbaselineskip
869 \sloppy
870 \let\\\@normalcr
871 }

```

`\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`.

```

872 \newcommand\caption@boxrestore@mini{%
873 \let\par\@par
874 \parindent\z@ \parskip\z@skip
875 \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).

```

876 \newcommand*\caption@normalsize{%
877 \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.

```
878 \let\caption@setfloatcapt\@firstofone
```

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

```

\caption@makeanchor 879 \newcommand*\caption@makecurrent[2]{}
\caption@start      880 \let\caption@makeanchor\@firstofone
\caption@@start     881 \let\caption@start\relax
\caption@freezeHref 882 \let\caption@@start\relax
\caption@defrostHref 883 \let\caption@freezeHref\relax
                    884 \let\caption@defrostHref\relax

```

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

```

885 \newcommand\caption@getttitle[1]{%
886 \caption@ifundefined\NR@getttitle
887 {\def\@currentlabelname{#1}}%
888 {\NR@getttitle{#1}}}

```

14 Support for sub-captions

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

```

889 \def\caption@DeclareSubType sub#1\@nil{%
890 \caption@Debug{Initializing subtype for `#1'\@gobble}%
891 \@namedef{caption@c@#1}{0}%
892 \@namedef{caption@beginsub#1}{\caption@beginsubfloat{#1}}
893 \@onlypreamble\caption@DeclareSubType

```

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

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

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

```
895 \caption@AtBeginDocument*{%
896   \caption@ifundefined\sf@counterlist{}{%
897     \@for\sf@temp:=\sf@counterlist\do{%
898       \expandafter\caption@DeclareSubType\sf@temp\@nil}}}
```

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

```
899 \newcommand*\caption@subtypehook{%
900   \ifx\caption\caption@subcaption \else
901     \caption@warmup
902     \caption@ifrefstepcounter{}{%
903       % no \caption or \subcaption in this (floating) environment yet
904       \caption@Debug{Increment \@cctype\ counter =\the\value\@cctype}%
905       \caption@l@stepcounter\@cctype
906       \let\caption@@addcontentsline\caption@addsubcontentsline}%
907     \ifnum\csname caption@c@\@cctype\endcsname=\value\@cctype \else
908       \caption@Debug{Reset sub\@cctype\ counter}%
909       \expandafter\xdef\csname caption@c@\@cctype\endcsname{%
910         \the\value\@cctype}%
911       \@stpel\@subcaption
912     \fi
913     \c@ContinuedFloat=0\relax
914     \let\caption@resetContinuedFloat\@gobble
915     \let\caption@addcontentsline\caption@kernel@addcontentsline
916     \let\caption@setfloatcapt\@firstofone
917     \caption@clearmargin
918     \caption@iflist{}{\let\caption@setlist\@gobble}%
919     \caption@setoptions{sub}%
920     \caption@setoptions{subfloat}% for subfig-package compatibility
921     \let\caption\caption@subcaption
922     \let\phantomcaption\caption@subphantom
923     \let\makecaption\caption@makecaption
924   \fi}%

```

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

```
925 \newcommand*\caption@subcaption{%
926   \caption@checkgrouplevel{sub}\subcaption
927   \caption@star
928   {\caption@refstepcounter\@subcaption}%
929   {\caption@dblarg{\@caption\@subcaption}}}
```

`\caption@subphantom` Same as `\phantomcaption`, but for sub-captions.

```
930 \newcommand*\caption@subphantom{%
931   \caption@checkgrouplevel{sub}\phantomsubcaption
932   \caption@refstepcounter\@subcaption}
```

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

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

```
933 \let\caption@kernel@addcontentsline\caption@addcontentsline
934 \renewcommand*\caption@addcontentsline[2]{%
```

```

935 \sf@ifpositiontop{\@listsubcaptions{#1}}{}%
936 \caption@kernel@addcontentsline{#1}{#2}%
937 \sf@ifpositiontop{}{\@listsubcaptions{#1}}%
938 \caption@addsubcontentslines{#1}}

939 \newcommand*\caption@addsubcontentslines[1]{%
940 \begingroup
941 \caption@subcontentslines
942 \endgroup
943 \caption@clearsubcontentslines}%

caption@addsubcontentsline Add a pending sub-caption list entry.
944 \newcommand*\caption@addsubcontentsline[4]{%
945 \begingroup
946 \let\label\@gobble \let\index\@gobble \let\glossary\@gobble
947 \protected@edef\@tempa{\endgroup
948 \noexpand\g@addto@macro\noexpand\caption@subcontentslines{%
949 \noexpand\@namedef{the#2}{\csname the#2\endcsname}%
950 \ifx\@currentHref\@undefined \else
951 \noexpand\def\noexpand\@currentHref{\@currentHref}%
952 \fi
953 \protect\caption@addcontentsline{#1}{#2}{#3}{#4}}}%
954 \@tempa}

ion@checksubcontentslines Checks if the list of pending sub-captions is empty, if not, a warning will be issued.
955 \newcommand*\caption@checksubcontentslines{%
956 \ifx\caption@subcontentslines\@empty \else
957 \caption@Error{%
958 Something's wrong--perhaps a missing \protect\caption\MessageBreak
959 in the last figure or table}%
960 \caption@clearsubcontentslines
961 \fi}

ion@clearsubcontentslines Clear pending sub-caption list entries.
962 \newcommand*\caption@clearsubcontentslines{%
963 \global\let\caption@subcontentslines\@empty}

964 \caption@AtBeginDocument*{%
965 \caption@ifundefined\sf@ifpositiontop{\let\sf@ifpositiontop\@gobbletwo}}%
966 \caption@clearsubcontentslines
967 \g@addto@macro\caption@typehook{\caption@checksubcontentslines}%
968 \AtEndDocument{\caption@checksubcontentslines}}%

```

15 Document class & Babel package support

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

```

969 \caption@ifundefined\smf@makecaption{}{\let\smf@makecaption\@makecaption}

```

15.2 The beamer class

```

970 \@ifclassloaded{beamer}{%
971 \caption@InfoNoLine{beamer document class}%

```

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

```

972 \expandafter\let\expandafter\caption@ORI@figure
973 \csname\string\figure\endcsname
974 \@namedef{\string\figure}[#1]{%
975   \caption@ORI@figure[#1]%
976   \caption@settype{figure}}
977 \expandafter\let\expandafter\caption@ORI@table
978 \csname\string\table\endcsname
979 \@namedef{\string\table}[#1]{%
980   \caption@ORI@table[#1]%
981   \caption@settype{table}}
982 {}

```

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.

```

983 \caption@ifundefined\scr@caption{}{%
984   \caption@AtBeginDocument{\let\scr@caption\caption}}

```

15.4 The frenchb Babel option

Suppress “Package frenchb.1df Warning: The definition of `\@makecaption` has been changed, frenchb will NOT customize it.” (but only if we emulate this customization)

```

985 \@nameuse{caption@frenchb}\@nameundef{caption@frenchb}

```

15.5 The frenchle/pro package

```

986 \caption@AtBeginDocument{\caption@ifundefined\frenchTeXmods{}{%
987   \caption@InfoNoLine{frenchle/pro package is loaded}%
988   \let\captionfont@ORI\captionfont
989   \let\captionlabelfont@ORI\captionlabelfont
990   \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`.

```

991 \caption@ifundefined\GOfrench
992   {\let\caption@tempa\@firstofone}%
993   {\def\caption@tempa{\g@addto@macro\GOfrench}}%
994 \caption@tempa{%
995   \let\captionfont\captionfont@ORI
996   \let\captionfont@ORI\@undefined
997   \let\captionlabelfont\captionlabelfont@ORI
998   \let\captionlabelfont@ORI\@undefined
999   \let\@makecaption\@makecaption@ORI
1000  \let\@makecaption@ORI\@undefined

```

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

```

1001   \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.

1002 \let\caption@tcORI\@tablescaption
1003 \def\@tablescaption{\caption@star\relax\caption@tcORI}%

\ffrench \ffrench and \tfrench reflect \fnum@figure and \fnum@table when
\tfrench 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.

1004 \let\@eatDP\@undefined
1005 \let\caption@tempa\@empty
1006 \ifx\ffrench\figure
1007 \l@addto@macro\caption@tempa{\let\fnum@figure\ffrench}%
1008 \fi
1009 \ifx\tfrench\table
1010 \l@addto@macro\caption@tempa{\let\fnum@table\tfrench}%
1011 \fi
1012 \def\ffrench{\ifx\listoffigures\relax\else\figurename~\thefigure\fi}%
1013 \def\tfrench{\ifx\listoftables\relax\else\tablename~\thetable\fi}%
1014 \caption@tempa

1015 }%
1016 }}

```

15.6 The hungarian and magyar Babel option

```

1017 \def\caption@tempa#1{%
1018 \ifundefined{extras#1}\caption@AtBeginDocument\@firstofone{%
1019 \ifundefined{extras#1}}{%
1020 \caption@InfoNoLine{#1 babel option is loaded}%
1021 \expandafter\addto\csname extras#1\endcsname{%
1022 % reverse changes made by magyar.ldf
1023 \let\makecaption\caption@makecaption
1024 \babel@save\@makecaption
1025 \caption@redefine
1026 \babel@save\@caption}%
1027 }}
1028 \caption@tempa{hungarian}%
1029 \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.

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

1033 \long\def\caption@ifpackageloaded#1[#2]#3#4{%
1034 \ifpackageloaded{#1}\@firstofone{%

```

```

1035 \caption@Debug{#1 package is not loaded (yet)\@gobble}%
1036 \caption@AtBeginDocument}{%
1037 \caption@ifpackageloaded{#1}{#2}{#3}{#4}}}%
1038 \@onlypreamble\caption@@IfPackageLoaded

1039 \newcommand\caption@ifpackageloaded[1]{%
1040 \@testopt{\caption@ifpackageloaded{#1}}{}}}%
1041 \@onlypreamble\caption@ifpackageloaded

1042 \long\def\caption@@ifpackageloaded#1[#2]{%
1043 \@ifpackageloaded{#1}{%
1044 \caption@InfoNoLine{#1 package is loaded}%
1045 \@ifpackagelater{#1}{#2}\@firstoftwo{%
1046 \caption@Error{%
1047 For a successful cooperation we need at least version\MessageBreak
1048 `#2' of package #1,\MessageBreak
1049 but only version\MessageBreak
1050 '\csname ver@#1.\@pkgextension\endcsname'\MessageBreak
1051 is available}%
1052 \@secondoftwo}%
1053 }{\@secondoftwo}}}%
1054 \@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.

```

1055 \newcommand*\caption@clearmargin{%
1056 \setcaptionmargin\z@
1057 \let\caption@minmargin\@undefined}

1058 \caption@setbool{needfreeze}{0}
1059 \caption@AtBeginDocument*{%
1060 \caption@ifneedfreeze{%

```

`\caption@freeze` `\caption@freeze`

Used by the `fitpage` & `sidecap` package support.

```

1061 \newcommand*\caption@freeze{%
1062 \global\let\caption@SCcontinued\relax
1063 \global\let\caption@SCsetup\@undefined
1064 \global\let\caption@SClentry\@undefined
1065 \global\let\caption@SCtext\@undefined
1066 \global\let\caption@SClabel\@undefined

1067 \let\caption@frozen@ContinuedFloat\ContinuedFloat
1068 \def\ContinuedFloat{%
1069 \caption@withoptargs\caption@SC@ContinuedFloat}%
1070 \def\caption@SC@ContinuedFloat##1{%
1071 \let\caption@frozen@setcounter\setcounter
1072 \let\caption@frozen@addtocounter\addtocounter
1073 \def\setcounter####1####2{\csname c@####1\endcsname####2\relax}%
1074 \def\addtocounter####1####2{\advance\csname c@####1\endcsname ####2\relax}%
1075 \caption@frozen@ContinuedFloat##1%
1076 \global\let\caption@SCcontinued\caption@frozen@ContinuedFloat
1077 \let\setcounter\caption@frozen@setcounter
1078 \let\addtocounter\caption@frozen@addtocounter}%
1079 \let\caption@frozen@setup\caption@setup
1080 \def\caption@setup##1{%

```

```

1081     \caption@g@addto@list\caption@SCsetup{##1}%
1082     \caption@frozen@setup{##1}}%
1083 \let\caption@frozen\caption
1084 \def\caption{%
1085     \def\caption{\caption@Error{%
1086         Only one \noexpand\caption can be placed in this environment}}%
1087     \let\captionsetup\caption@setup
1088     \let\caption@@refstepcounter\caption@l@stepcounter
1089     \caption@frozen}%
1090 \let\caption@frozen@\caption
1091 \long\def\@caption##1[##2]##3{%
1092 %    \@bsphack
1093     \gdef\caption@SClentry{##2}%
1094     \gdef\caption@SCtext{##3}%
1095     \let\@currentlabel\caption@SCcurrentlabel
1096 %    \@esphack}%
1097     \ignorespaces}%
1098 \let\caption@frozen@label\label
1099 \def\label##1{%
1100     \ifx\@currentlabel\caption@SCcurrentlabel
1101         \@bsphack\gdef\caption@SClabel{##1}\@esphack
1102     \else
1103         \caption@frozen@label{##1}%
1104     \fi}%
1105 \def\caption@SCcurrentlabel{???}%
1106 }%

\caption@warmup \caption@warmup
1107 \newcommand*\caption@warmup{%
1108     \ifx\caption@frozen\@undefined\else
1109         \let\caption@setup\caption@frozen@setup
1110         \let\caption\caption@frozen
1111         \let\@caption\caption@frozen@
1112         \let\label\caption@frozen@label
1113     \fi}

\caption@defrost \caption@defrost
1114 \newcommand*\caption@defrost{%
1115     \ifx\caption@frozen\@undefined
1116         \caption@defrost@setup
1117         \ifx\caption@SCtext\@undefined \else
1118             \expandafter\expandafter\expandafter\caption
1119             \expandafter\expandafter\expandafter[%
1120             \expandafter\expandafter\expandafter{%
1121             \expandafter\caption@SClentry\expandafter}\expandafter]%
1122             \expandafter{\caption@SCtext}%
1123         \fi
1124         \ifx\caption@SClabel\@undefined \else
1125             \expandafter\label\expandafter{\caption@SClabel}%
1126         \fi
1127     \else
1128         \caption@Error{Internal Error:\MessageBreak
1129             \noexpand\caption@defrost in same group as \string\caption@freeze}%
1130     \fi}%

```



```

1131 \newcommand*\caption@defrost@setup{%
1132   \caption@SCcontinued
1133   \ifx\caption@SCsetup\@undefined \else
1134     \expandafter\captionsetup\expandafter{\caption@SCsetup}%
1135   \fi}%

1136 }{\let\caption@warmup\relax}%
1137 \caption@undefbool{needfreeze}}

```

16.1 The float package

The float package usually do not use the L^AT_EX 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.)

```

1138 \caption@ifpackageloaded{float}[2001/11/08 v1.3d]{%
1139   \@ifpackageloaded{floatrow}{%
1140     \caption@ifpackageloaded{floatrow}[2007/08/24 v0.2a]{}{}%
1141   }%

```

`\@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>]`.

```

1142 \let\caption@ORI@float@setevery\@float@setevery
1143 \def\@float@setevery#1{%
1144   \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.

```

1145   \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.

```

1146   \renewcommand\caption@setfloatcapt[1]{%
1147     \let\@makecaption\caption@make
1148     \global\setbox\@floatcapt\vbox{%
1149       \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.

```

1150     \float@getstyle\float@style{#1}%
1151     \caption@setstyle*\float@style
1152     \caption@setoptions\float@style
1153   }{}%
1154   \caption@freezeHref % will be defrosted in \float@makebox
1155   \caption@ORI@float@setevery{#1}}%

```

`\caption@typehook` L^AT_EX and almost every other packages use `\langle type \rangle` 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.

```

1156   \g@addto@macro\caption@typehook{%
1157     \expandafter\ifx\csname #1name\endcsname\relax
1158       \expandafter\let\csname #1name\endcsname\expandafter\endcsname
1159       \csname fname@#1\endcsname
1160     \fi}%

```

`\fs@plaintop` 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.

```

1161   \g@addto@macro\fs@plaintop{\def\@fs@mid{\vspace\abovecaptionskip\relax}}%
1162   \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).

```

1163   \providecommand*\float@ifstyle[1]{%
1164     \expandafter\ifx\csname fst@#1\endcsname\relax
1165       \expandafter\@secondoftwo
1166     \else
1167       \expandafter\@firstoftwo
1168     \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.

```
1169 \providecommand*\float@getstyle[2]{%
1170   \edef#1{%
1171     \noexpand\expandafter\noexpand\@gobblefour\noexpand\string
1172     \expandafter\expandafter\expandafter\noexpand
1173     \csname fst@#2\endcsname}%
1174   \edef#1{#1}%
1175   \caption@Debug{floatstyle{#2} = '#1'}}%
```

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

```
1176 \providecommand*\float@setstyle[2]{%
1177   \expandafter\edef\csname fst@#1\endcsname{%
1178     \expandafter\noexpand\csname fs@#2\endcsname}}%
```

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

```
1179 \providecommand*\float@dostyle[1]{%
1180   \@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.)

```
1181 \providecommand*\float@ifcaption[1]{%
1182   \expandafter\ifx\csname @float@c@#1\endcsname\float@caption
1183   \expandafter\@firstoftwo
1184   \else
1185   \expandafter\@secondoftwo
1186   \fi}%

```

```
1187 } }{%
1188   \providecommand*\float@ifstyle[1]{\@secondoftwo}%
1189   \providecommand*\float@ifcaption[1]{\@secondoftwo}%
1190 % \clearcaptionsetup{boxed}% used by the floatrow package?
1191 }
```

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

```
1192 \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’.

```
1193 \caption@ifbool{ruled}{%
1194   \captionsetup[ruled]{margin=0pt,minmargin=0,slc=0}%

```

```

1195 }{%
1196   \DeclareCaptionStyle{ruled}{labelfont=bf,labelsep=space,strut=0}%
1197 }
1198 \caption@undefbool{ruled}

```

16.2 The floatflt package

```

1199 \caption@IfPackageLoaded{floatflt}[1996/02/27 v1.3]{%
\floatingfigure We patch \floatingfigure so \caption@floatflt will be used.
1200   \let\caption@ORI@floatingfigure\floatingfigure
1201   \def\floatingfigure{%
1202     \caption@floatflt{figure}%
1203     \caption@ORI@floatingfigure}%
\floatingtable Same with \floatingtable...
1204   \let\caption@ORI@floatingtable\floatingtable
1205   \def\floatingtable{%
1206     \caption@floatflt{table}%
1207   %   \caption@setautoposition b%
1208     \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.
1209   \newcommand*\caption@floatflt[1]{%
1210     \caption@settype{#1}%
1211     \caption@clearmargin
1212     \caption@setfullparboxrestore
1213     \caption@setoptions{floating#1}}%
1214 }{}

```

16.3 The fltpage package

```

1215 \caption@IfPackageLoaded{fltpage}[1998/10/29 v.0.3]{%
1216   \caption@setbool{needfreeze}{1}%
\FP@helpNote Original code:
    \newcommand{\FP@helpNote}[2]{%
      \typeout{FP#1 is inserted on page \pageref{#2}!}}%

1217   \renewcommand\FP@helpNote[2]{%
1218     \begingroup % save \caption@thepage
1219       \caption@pageref{#2}%
1220       \typeout{FP#1 is inserted on page \caption@thepage!}%
1221     \endgroup}%

```

\FP@floatBegin **Original code:**

```

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

1222 \renewcommand*\FP@floatBegin[1]{%
1223   \def\@captype{#1}%
1224   \let\FP@LabelText\@empty
1225   \begin{lrbox}{\FP@floatCorpusBOX}%
1226   \caption@freeze
1227   \caption@ifFPrefcap
1228   {}%
1229   {\def\label##1{\@bsphack\gdef\FP@LabelText{##1}\@esphack}}%
1230   \ignorespaces}%

```

\FP@floatEnd **Original code:**

```

\newcommand{\FP@floatEnd}{%
  \end{lrbox}%
  \global\setbox\FP@floatCorpusBOX=\box\FP@floatCorpusBOX
  \stepcounter{FP@\@captype C}%
  \FP@savedLabelCommand{\FP@positionLabel}%
  \FP@helpNote{\@captype}{\FP@positionLabel}%
  \FP@float
    {\FP@positionLabel}% location label test
  {\begin{\@captype}[p!]}
    \usebox{\FP@floatCorpusBOX}%
    \refstepcounter{\@captype}%
    \ifthenelse{\equal{\FP@LabelText}{\@empty}}{
      {}{\FP@savedLabelCommand{\expandafter\protect\FP@LabelText}}}%
    \end{\@captype}}
  {\addtocounter{\@captype}{-1}}
  {\begin{\@captype}[b!]}
    \ifthenelse{\equal{\FP@guide}{\@empty}}{
      {}{\ifthenelse{\equal{\@captype}{figure}}{
        {\renewcommand{\fnun@figure}{\old@Fnum\ {\FP@guide}}}%
        {\renewcommand{\fnun@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{\@capttype}}%
}%

1231 \renewcommand*\FP@floatEnd{%
1232     \end{lrbox}%
1233     \stepcounter{FP@\@capttype C}%
1234     \caption@label\FP@positionLabel
1235     \FP@helpNote\@capttype\FP@positionLabel
1236     \edef\FP@RestoreCounter{%
1237         \noexpand\setcounter{\@capttype}{\the\value\@capttype}%
1238         \noexpand\setcounter{ContinuedFloat}{\the\value{ContinuedFloat}}}%
1239     \FP@float
1240     {\FP@positionLabel}% location label test
1241     {\begin\@capttype[p!]}%
1242         \usebox\FP@floatCorpusBOX
1243         \caption@defrost@setup
1244         \caption@ifFPlistcap
1245             {\caption@refstepcounter\@capttype
1246             \expandafter\caption@makecurrent\expandafter\@capttype
1247                 \expandafter{\caption@SCLentry}}}%
1248             {\expandafter\captionlistentry\expandafter{\caption@SCLentry}}}%
1249         \caption@makeanchor\relax
1250         \ifx\FP@LabelText\@empty \else
1251             \expandafter\label\expandafter{\FP@LabelText}%
1252         \fi
1253     \end\@capttype}%
1254     {\FP@RestoreCounter
1255     \@ifundefined{theH\@capttype}{}%
1256     \expandafter\l@addto@macro\csname theH\@capttype\endcsname{.FP}}}%
1257     {\begin\@capttype[b!]}%
1258         \let\FP@savedSetfnumCommand\caption@setfnum
1259         \def\caption@setfnum##1{%
1260             \FP@savedSetfnumCommand{##1}%
1261             \ifx\FP@guide\@empty \else
1262                 \expandafter\l@addto@macro\csname fnum@##1\endcsname{\ \FP@guide}}%
1263             \fi}%
1264         \setlength\abovecaptionskip{2pt plus 2pt minus 1pt}% length above captio
1265         \setlength\belowcaptionskip{2pt plus 2pt minus 1pt}% length below captio
1266         \caption@setoptions{FP\@capttype}%
1267         \FP@separatorCaption
1268         \caption@ifFPlistcap{}{\let\caption@addcontentsline\@gobbletwo}%
1269         \caption@defrost
1270     \end\@capttype}%
1271 }%

1272 \caption@For{typelist}{%
1273     \newcounter{FP@#1C}%
1274     \newenvironment{FP#1}{\FP@floatBegin{#1}}{\FP@floatEnd}}%
1275 }%
1276 \let\caption@ifFPlistcap\undefined

```

```

1277 \let\caption@ifFPrefcap\@undefined
1278 }

```

16.4 The hyperref package

```

1279 \caption@ifPackageLoaded{hyperref}[2003/11/30 v6.74m]{%
1280 % Test if hyperref has stopped early
1281 \caption@ifundefined\ifhyp@stoppedearly{%
1282   \newif\ifhyp@stoppedearly
1283   \caption@ifundefined\H@refstepcounter\hyp@stoppedearlytrue{%
1284     \caption@ifundefined\hyper@makecurrent\hyp@stoppedearlytrue{%
1285       \caption@ifundefined\measuring@true\hyp@stoppedearlytrue{}}}{}}%
1286 \ifhyp@stoppedearly % hyperref has stopped early
1287   \caption@InfoNoLine{%
1288     Hyperref support is turned off\MessageBreak
1289     because hyperref has stopped early}%
1290 \else
1291 %
1292 \g@addto@macro\caption@prepareslc{\measuring@true}%

```

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

```

1293 \renewcommand*\caption@@refstepcounter{\H@refstepcounter}%

```

We redefine \caption@makecurrent so a hyperref label will be defined inside \@caption.

Note: Will be redefined by \caption@start.

```

1294 \renewcommand*\caption@makecurrent[2]{%
1295   \caption@makecurrentHref{#1}%
1296   \caption@Debug{hyperref current=\@currentHref}%
1297   \caption@getttitle{#2}}%
1298 \newcommand*\caption@makecurrentHref{\hyper@makecurrent}%

```

We redefine \caption@makeanchor so a hyperref anchor will be set inside \@caption.

Note: Will be redefined by \caption@start.

```

1299 \renewcommand\caption@makeanchor[1]{%
1300   \caption@Debug{hyperref anchor: \@currentHref}%
1301   % If we cannot have nesting, the anchor is empty.
1302   \ifHy@nesting
1303     \expandafter\hyper@@anchor\expandafter{\@currentHref}{#1}%
1304   \else
1305     \Hy@raisedlink{%
1306       \expandafter\hyper@@anchor\expandafter{\@currentHref}{\relax}%
1307       }#1%
1308   \fi}%
1309 \g@addto@macro\caption@prepareslc{\let\caption@makeanchor\@firstofone}%

```

The hycap option

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

```

1310 \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`.

```
1311 \def\caption@start{\caption@ifhypcap\caption@start@relax}%
1312 \def\caption@start@{%
```

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

```
1313 \caption@makestart\@captype
1314 \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.)

```
1315 \global\@capstarttrue
1316 \let\hc@currentHref\@currentHref
1317 \def\caption@makecurrentHref##1{%
1318 \global\@capstartfalse
1319 \global\let\@currentHref\hc@currentHref}%
```

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

```
1320 \let\caption@makeanchor\@firstofone
1321 }%
```

`\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 `hypertexnames=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 `hypertexnames=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.

```
1322 \newcommand*\caption@makestart[1]{%
1323 \begingroup
1324 \Hy@hypertexnamesfalse
1325 % \gdef\@currentHlabel{}%
1326 \hyper@makecurrent{#1.caption}%
1327 \endgroup
1328 \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.

```

1329 \newcommand*\caption@startanchor[1]{%
1330   \ifvmode\begin{group}
1331     \caption@Debug{hypcap anchor: #1 (vertical mode)}%
1332     \@tempdima\prevdepth
1333     \nointerlineskip
1334     \vspace*{-\caption@hypcapspace}%
1335     \caption@anchor{#1}%
1336     \vspace*{\caption@hypcapspace}%
1337     \prevdepth\@tempdima
1338   \endgroup\else
1339     \caption@Debug{hypcap anchor: #1 (horizontal mode)}%
1340     \caption@anchor{#1}%
1341   \fi}%

```

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

```

1342 \newcommand*\caption@anchor[1]{%
1343   \ifmeasuring@ \else
1344     \caption@raisedlink{\hyper@anchorstart{#1}\hyper@anchorend}%
1345   \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`.)

```

1346 \ifx\HyperRaiseLinkLength\@tempdima
1347   \def\caption@raisedlink#1{\ifvmode#1\else\Hy@raisedlink{#1}\fi}%
1348 \else
1349   \let\caption@raisedlink\Hy@raisedlink
1350 \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.

```

1351 \def\caption@@start{%
1352   \caption@ifundefined\hc@currentHref{%
1353     \caption@Warning{%
1354       The option ‘hypcap=true’ will be ignored for this\MessageBreak
1355       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`.

```

1356 \def\caption@freezeHref{%
1357   \let\caption@ORI@start\caption@start
1358   \def\caption@start{\let\caption@start\caption@ORI@start}%
1359 %
1360 %
1361 %
1362 \global\let\caption@currentHref\undefined
1363 \def\caption@@start{\global\let\caption@currentHref\@currentHref}%

```

```

1364      \let\caption@ORI@setfloatcapt\caption@setfloatcapt
1365      \renewcommand*\caption@setfloatcapt{%
1366        \ifx\caption@currentHref\@undefined \else
1367          \let\caption@makeanchor\@firstofone
1368        \fi
1369        \caption@ORI@setfloatcapt}}%

\caption@defrostHref If there is a freezed \@currentHref, we set the hyperref anchor here.
1370      \def\caption@defrostHref{%
1371        \ifx\caption@currentHref\@undefined \else
1372          \caption@startanchor\caption@currentHref
1373          \global\let\caption@currentHref\@undefined
1374        \fi}%

\float@makebox Do our own redefinition of \float@makebox, if it was redefined by the hyperref pack-
age.
1375      \caption@ifundefined\HyOrg@float@makebox{}{%
1376        \caption@Debug{%
1377          Redefining \noexpand\float@makebox (again)\@gobble}%
1378        \let\caption@ORI@float@makebox\float@makebox % save for compatibility mode
1379        \renewcommand\float@makebox[1]{%
1380          \HyOrg@float@makebox{#1}\relax \caption@defrostHref}}%
1381      }%

1382    \fi}{}

```

16.5 The hypcap package

```

1383 \caption@ifPackageLoaded{hypcap}{% v1.0
1384   \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 environ-
ment, at least unless a new \capstart will be placed.
1385     \let\caption@ORI@capstart\capstart % save for compatibility mode
1386     \caption@ifundefined\capstarttrue % check for v1.10 of hypcap package
1387     {\def\capstart{\caption@start@}}%
1388     {\def\capstart{\ifcapstart\caption@start@\fi}}%
1389     \let\caption@start\relax
1390     \let\caption@@start\relax

\caption@hypcapspace Furthermore we map our \caption@hypcapspace to \hypcapspace offered by
the hypcap package.
1391     \caption@set@bool\caption@ifhypcap 1%
1392     \renewcommand*\caption@hypcapspace{\hypcapspace}%

1393   \fi}{}

```

16.6 The listings package

1394 \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.

```
1395 \let\caption@ORI@lst@MakeCaption\lst@MakeCaption
1396 \def\lst@MakeCaption#1{% #1 is ‘t’ or ‘b’
1397   \begingroup
```

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

```
1398   \ifdim\hsize>\linewidth
1399     \hsize\linewidth
1400   \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.)

```
1401   \caption@setposition{#1}%
1402   \caption@iftop{%
1403     \@tempdima\belowcaptionskip
1404     \belowcaptionskip\abovecaptionskip
1405     \abovecaptionskip\@tempdima}{}%
```

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

```
1406   \caption@setup{rule=0}%
```

Afterwards we set the local ‘lstlisting’ options.

```
1407   \caption@setoptions{lstlisting}%
```

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

```
1408   \caption@setautoposition{#1}%
```

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

```
1409   \caption@begin{lstlisting}%
1410   \caption@ORI@lst@MakeCaption{#1}%
1411   \caption@end
1412   \endgroup}%
```

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

```
\lst@maketitle 1413 \def\lst@makecaption{\caption@starfalse\@makecaption}%
1414 \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.

```
1415 \providecommand*\ext@lstlisting{lol}%
```

```
1416 }{ }
```

16.7 The longtable package

```

\LTcapttype \LTcapttype is preset to table.
1417 \providecommand*\LTcapttype{table}

1418 \caption@ifPackageLoaded{longtable}[1995/05/24 v3.14]{%
1419   \RequirePackage{ltcaption}[2007/09/01]%
1420   \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.

1421 \caption@AtBeginDocument{%
1422   \let\caption@ORI@LT@array\LT@array
1423   \renewcommand*\LT@array{%

\captionsetup for longtable:
1424   \global\let\caption@opt@@longtable\@undefined
1425   \def\captionsetup{%
1426     \noalign\bgroup
1427     \ifstar\captionsetup\captionsetup}% gobble *
1428   \def\captionsetup##1{\LT@captionsetup{##1}\egroup}%
1429   \def\LT@captionsetup##1{%
1430     \captionsetup@starttrue\caption@setup@options[@longtable]{##1}%
1431     \global\let\caption@opt@@longtable\caption@opt@@longtable}%

\captionabove & \captionbelow for longtable: (KOMA-Script document class)
1432   \def\captionabovetrue{\LT@captionsetup{position=t}}%
1433   \def\captionabovefalse{\LT@captionsetup{position=b}}%

\captionlistentry for longtable:
1434   \def\captionlistentry{%
1435     \noalign\bgroup
1436     \ifstar{\egroup\LT@captionlistentry}% gobble *
1437     {\egroup\LT@captionlistentry}}%
1438   \def\LT@captionlistentry##1{%
1439     \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

1440 %   \caption@ifhyperc{%
1441 %     \let\caption@ORI@hyper@makecurrent\hyper@makecurrent
1442 %     \def\hyper@makecurrent##1{%
1443 %       \let\hyper@makecurrent\caption@ORI@hyper@makecurrent
1444 %       \caption@makestart{##1}%
1445 %       \let\Hy@LT@currentHlabel\@currentHlabel
1446 %       \let\Hy@LT@currentHref\@currentHref
1447 %       \def\hyper@makecurrent###1{%
1448 %         \let\@currentHlabel\Hy@LT@currentHlabel
1449 %         \let\@currentHref\Hy@LT@currentHref}}%
1450 %       \let\caption@ORI@ContinuedFloat\ContinuedFloat

```

```

1451 %      \def\ContinuedFloat{\noalign{%
1452 %          \gdef\caption@setContinuedFloat{%
1453 %              \let\caption@resetContinuedFloat\@gobble}%
1454 %          \def\caption@setoptions###1{%
1455 %              \g@addto@macro\caption@setContinuedFloat{%
1456 %                  \caption@setoptions{###1}}}%
1457 %          \let\@capttype\LTcapttype
1458 %          \caption@ORI@ContinuedFloat}}%
1459 %      }{%
1460 %          \def\ContinuedFloat{\noalign{%
1461 %              \caption@Error{%
1462 %                  \noexpand\ContinuedFloat inside longtables\MessageBreak
1463 %                  is only available with 'hyccap=true'}}}%
1464 %      }%
1465 %      \global\let\caption@setContinuedFloat\@empty
1466 %      \def\ContinuedFloat{\noalign{%
1467 %          \caption@Error{\noexpand\ContinuedFloat outside float}}}%
1468 %      \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}:

```

1469 \long\def\LT@c@ption#1[#2]#3{%
1470     \LT@makecaption#1{\csname fnun@\LTcapttype\endcsname}{#3}%
1471     \LT@captionlistentry{#2}}%

```

\LT@makecaption \LT@makecaption{<cmd>}{<label>}{<text>}

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:

```

1472 \renewcommand\LT@makecaption[3]{%
1473     \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`.)

```
1474      \caption@settype*\LTcaptype
1475      \ifdim\LTcapwidth=4in \else
1476        \setcaptionwidth\LTcapwidth
1477      \fi
1478      \caption@setoptions{longtable}%
1479 %      \caption@setContinuedFloat
1480      \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!

```
1481      \caption@setautoposition{\ifcase\LT@rows t\else b\fi}%
```

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

```
1482      \caption@startrue#1\caption@starfalse
1483      \caption@resetContinuedFloat\LTcaptype
1484      \caption@begin\LTcaptype
1485      \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...)

```
1486      \vskip-\ht\strutbox
```

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

```
1487      \caption@iftop{\vskip\belowcaptionskip}{\vskip\abovecaptionskip}%
1488      \caption@@make{#2}{#3}\endgraf
1489      \caption@iftop{\vskip\abovecaptionskip}{\vskip\belowcaptionskip}%
1490      \caption@end}%%
```

```
1491 }{ }
```

16.8 The picinpar package

```
1492 \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.

```
1493   \long\def\figwindow[#1,#2,#3,#4] {%
1494     \caption@window{figure}%
1495     \caption@setoptions{figwindow}%
1496     \begin{window}[#1,#2,{#3},\caption@wincaption{#4}] }%
```

```

1497 \long\def\tabwindow[#1,#2,#3,#4] {%
1498   \caption@window{table}%
1499   \caption@setoptions{tabwindow}%
1500   \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.**

```

1501 \newcommand*\caption@window[1]{%
1502   \let\@makecaption\caption@@make
1503   \caption@setautoposition b%
1504   \caption@settype{#1}%
1505   \caption@clearmargin
1506   \caption@setfullparboxrestore}%

```

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

```

1507 \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`.

```

1508   \ifdim\picwd=\z@
1509     \let\caption@makecurrent\@gobbletwo
1510     \let\caption@@start\relax
1511     \caption@prepareslc
1512   \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^{OF}] {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.

```

1513   \edef\@tempa{\expandafter\noexpand\@car#1\@nil}%
1514   \if\@tempa*%
1515     \let\@tempa\@firstofone
1516   \else\if\@tempa[%
1517     \let\@tempa\@firstofone
1518   \else
1519     \let\@tempa\@empty
1520   \fi\fi
1521   \expandafter\caption\@tempa{#1}}%

```

```

1522 }{}

```

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 `\@capytype`, 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.

```

1523 \newcommand*\piccaptiontype[1]{\def\@piccapytype{#1}}

```

```

1524 \caption@ifpackageloaded{picins}{%

```

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

```
1525 \caption@ifundefined\@piccaptiontype{%
1526   \caption@iftype{%
1527     \let\@piccaptiontype\@capttype
1528   }{%
1529     \def\@piccaptiontype{figure}%
1530   }%
1531 }{}%
1532 \let\@capttype\@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.

```
1533 \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. `\@capttype` 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.

```
1534 \let\caption@ORI@make@piccaption\make@piccaption
1535 \def\make@piccaption{%
1536   \let\caption@ORI\caption
1537   \long\def\caption[##1]##2{%
1538     \caption@freezeHref % will be defrosted in \ivparpic
1539     \caption@settype\@piccaptiontype
1540 % \ifnum\c@piccaptionpos>2\relax
1541     \caption@clearmargin
1542 % \else
1543 % \captionwidth\z@ % do not use "width=" setting
1544 % \fi
1545     \caption@setfullparboxrestore
1546     \caption@setoptions{parpic}%
1547     \caption@setautoposition b%
```



```

1548     \expandafter\expandafter\expandafter\caption@ORI
1549     \expandafter\expandafter\expandafter[%
1550     \expandafter\expandafter\expandafter{%
1551     \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

1552     \caption@ORI@make@piccaption
1553     \let\caption\caption@ORI}%

```

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

```

1554     \let\caption@ORI@ivparpic\ivparpic
1555     \def\ivparpic(#1,#2)(#3,#4)[#5][#6]#7{%
1556     \let\caption@ORI@noindent\noindent
1557     \def\noindent{%
1558     \caption@defrostHref
1559     \let\noindent\caption@ORI@noindent
1560     \noindent}%
1561     \caption@ORI@ivparpic(#1,#2)(#3,#4)[#5][#6]{#7}%
1562     \let\noindent\caption@ORI@noindent}%

1563 }{%
1564 \let\piccaptiontype\@undefined
1565 }

```

16.10 The rotating package

```

1566 \caption@ifPackageLoaded{rotating}[1995/08/22 v2.10]{%

```

`\rotcaption` Make `\rotcaption*` work.

```

1567 \def\rotcaption{\let\makecaption\@makerotcaption\caption}%
1568 % \let\@rotcaption\@undefined

```

`\rotcaptionof` Make `\rotcaptionof(*)` work.

```

1569 \def\rotcaptionof{%
1570 \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
}

```

```

\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.)

```

1571 \long\def\@makerotcaption#1#2{%
1572     \ifdim\captionwidth=\z@
1573     \setcaptionwidth{.8\textheight}%
1574     \caption@slc{#1}{#2}{.8\vsizer}%
1575     \let\caption@makerot\caption@@make
1576     \caption@clearmargin
1577 %     \long\def\caption@parbox##1##2{\hbox{\hsize=.8\textheight\relax##2}}%
1578 %     (not needed because \rotatebox uses an \hbox anyway)
1579     \let\caption@parbox\@secondoftwo}%
1580     \caption@set@bool\caption@ifslc0% been there, done that
1581 \fi

1582 \rotatebox{90}{\caption@makerot{#1}{#2}}%
1583 \nobreak\hspace{12pt}}%

1584 \newcommand\caption@makerot[2]{%
1585     \begin{minipage}\captionwidth\caption@@make{#1}{#2}\end{minipage}}%

1586 \caption@For{typelist}{%
1587     \newenvironment{sideways#1}{\@rotfloat{#1}}{\end@rotfloat}%
1588     \newenvironment{sideways#1*}{\@rotdblfloat{#1}}{\end@rotdblfloat}}%
1589 {}

```

16.11 The sidecap package

```

1590 \caption@IfPackageLoaded{sidecap}[1999/05/11 v1.4d]{%
1591     \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.)

```

1592 \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`.

```

1593 \let\caption@ORI@SC@zfloat\SC@zfloat
1594 \def\SC@zfloat#1#2#3[#4]{%

```

First we use the original definition, but restore `\caption` and `\label` so `\caption@freeze` and `\caption@warmup` will work correctly.

```

1595     \caption@ORI@SC@zfloat{#1}{#2}{#3}[#4]%
1596     \SC@RestoreCommands

```

Since the `sidecap` package uses our `\caption` code outside the environment the regular `\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.

```

1597     \caption@settype*{#2}%
1598     \caption@freeze

```

The `sidecap` package uses `\ifx\label\SC@label` to test if it is just inside a `SC-figure` or not. So we redefine `\SC@label` here so this test will still work.

```
1599 \let\SC@label\label}%
1600 \providecommand*\SC@RestoreCommands{%
1601 \let\caption=\SC@orig@caption \let\label=\SC@orig@label}%
```

`\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.

```
1602 \let\caption@ORI@endSC@FLOAT\endSC@FLOAT
1603 \def\endSC@FLOAT{%
```

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

```
1604 \let\caption@ORI@settype\caption@settype
1605 \def\caption@settype##1{% will be done in \xfloat
1606 \caption@ORI@settype*{##1}% do not change \@currentlabel
1607 \caption@setSC@justify
1608 %%% \caption@setoptions{SCfloat}%
1609 \caption@setoptions{SC\@captive}%
1610 \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

```
1611 \def\caption@setSC@justify{%
1612 \caption@clearmargin
1613 \caption@ifundefined\SC@justify{}{%
1614 \ifx\SC@justify\@empty \else
1615 \let\caption@hj\SC@justify
1616 \let\SC@justify\@empty
1617 \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!

```
1618 \let\caption\SC@orig@caption
1619 \def\SC@orig@caption[##1]##2{\caption@defrost}%
```

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

```
1620 \caption@setSC@justify % for compatibility mode
1621 \caption@ORI@endSC@FLOAT}%
```

```
1622 \newcommand*\caption@For@SC[2]{%
1623 \def#1{b}% = \sidecaptionvpos{#2}{b} (v1.6)
1624 \newenvironment{SC#2}%
1625 {\SC@float[#1]{#2}}{\endSC@float}%
1626 \newenvironment{SC#2*}%
```

```

1627         {\SC@dblfloat[#1]{#2}}{\endSC@dblfloat}}}%
1628 \caption@For{typelist}{%
1629     \expandafter\caption@For@SC\csname SC@#1@vpos\endcsname{#1}}}%
1630 }{}

```

16.12 The subfigure package

```

1632 \caption@IfPackageLoaded{subfigure}[2002/01/23 v2.1]{%
\sf@ifpositiontop If the subfigure package is loaded, we map \sf@ifpositiontop to \iffiguresetopcap
resp. \iftabletopcap, so the subfigure v2.1 options figbotcap etc. will still work.
1633 \def\sf@ifpositiontop{%
1634     \ifx\@cuptype\@undefined
1635         \expandafter\@gobbletwo
1636     \else\ifx\@cuptype\relax
1637         \expandafter\expandafter\expandafter\@gobbletwo
1638     \else
1639         \expandafter\expandafter\expandafter\sf@if@position@top
1640     \fi\fi}
1641 \def\sf@if@position@top{%
1642     \ifundefined{if\@cuptype topcap}%
1643     {\@gobbletwo}%
1644     {\@nameuse{if\@cuptype topcap}%
1645         \expandafter\@firstoftwo
1646     \else
1647         \expandafter\@secondoftwo
1648     \fi}}
1649 }{}

```

16.13 The supertabular and xtab packages

```

1650 \caption@IfPackageLoaded{supertabular}[2002/07/19 v4.1e]{%
\tablecaption Make \topcaption* and \bottomcaption* work.
1651 \renewcommand*\tablecaption{%
1652     \caption@star
1653     {\refstepcounter{table}}}%
1654     {\caption@dblarg{\@xtablecaption}}}%
\@xtablecaption Make \nameref and \autoref work.
1655 \let\caption@ORI@xtablecaption\@xtablecaption
1656 \long\def\@xtablecaption[#1]#2{%
1657     \caption@gettitle{#2}%
1658     \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}

```

```

1659 \long\def\ST@caption#1[#2]#3{\par%
1660 \caption@settype*{#1}%
1661 \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.

```

1662 \def\caption@fixposition{%
1663 \caption@setposition{\if@topcaption t\else b\fi}}%
1664 \caption@beginex{#1}{#2}{#3}%
1665 \caption@boxrestore
1666 \caption@normalsize
1667 \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
1668 \caption@end}%

```

```

1669 }{}

```

```

1670 \caption@ifpackageloaded{xtab}[2000/04/09 v2.3]{%

```

\tablecaption **Make \topcaption* and \bottomcaption* work.**

```

1671 \renewcommand*\tablecaption{%
1672 \caption@star
1673 {\refstepcounter{table}}%
1674 {\caption@dblarg{\@xtablecaption}}}%

```

\@xtablecaption **Make \nameref and \autoref work.**

```

1675 \let\caption@ORI@xtablecaption\@xtablecaption
1676 \long\def\@xtablecaption[#1]#2{%
1677 \caption@getttitle{#2}%
1678 \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{%
\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}}

```

```

1679 \long\def\ST@caption#1[#2]#3{\par%
1680   \caption@settype*{#1}%
1681   \caption@setoptions{xtabular}%
1682   \def\caption@fixposition{%
1683     \caption@setposition{\if@topcaption t\else b\fi}}%
1684   \@initisotab
1685   \caption@beginex{#1}{#2}{#3}%
1686   \caption@boxrestore
1687   \caption@normalsize
1688   \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
1689   \caption@end
1690   \global\advance\ST@pageleft -\PWSTcapht
1691   \ST@trace\tw@{Added caption. Space left for xtabular: \the\ST@pageleft}}%
1692 }{}

```

16.14 The threeparttable package

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

```

1694 \let\caption@ORI@threeparttable\threeparttable
1695 \renewcommand*\threeparttable{%
1696   \caption@settype{table}%
1697   \caption@setposition a% ?
1698   \caption@clearmargin
1699   \caption@setoptions{threeparttable}%
1700   \caption@ORI@threeparttable}%

```

`\measuredfigure` Same here...

```

1701 \let\caption@ORI@measuredfigure\measuredfigure
1702 \renewcommand*\measuredfigure{%
1703   \caption@settype{figure}%
1704   \caption@setposition a% ?
1705   \caption@clearmargin
1706   \caption@setoptions{measuredfigure}%
1707   \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}

1708 \def\TPT@caption#1[#2]#3{%
1709   \gdef\TPT@docapt{%
1710     \global\let\TPT@docapt\@undefined
1711     \caption@setautoposition\caption@TPT@position
1712     \TPT@LA@caption{#1}[{#2}]{#3}}%
1713   \ifx\TPT@hsize\@empty

```

```

1714     \let\label\TPT@gatherlabel % Bug: does not work for measuredfigures
1715     \gdef\caption@TPT@position{t}%
1716     \g@addto@macro\TPT@docapt\caption@TPT@eatvskip
1717     \else
1718     \def\caption@TPT@position{b}%
1719     \TPT@docapt
1720     \fi
1721     \ignorespaces}%

1722 %\newcommand*\caption@TPT@eatvskip{\vskip-.2\baselineskip}%
1723 \def\caption@TPT@eatvskip#1\vskip{#1\@tempdima}%

1724 {}{}

```

16.15 The wrapfig package

```

1725 \caption@ifPackageLoaded{wrapfig}{% ver 3.3 (Oct 12, 1999)
\float@ifstyle \float@ifstyle{<type>}{<if-clause>}{<else-clause>}
(see float package support for details)
1726 \providecommand*\float@ifstyle[1]{%
1727     \expandafter\ifx\csname fst@#1\endcsname\relax
1728     \expandafter\@secondoftwo
1729     \else
1730     \expandafter\@firstoftwo
1731     \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.

```

1732 \newcommand*\caption@restylewrapfloat[1]{%
1733     \expandafter\ifx\csname caption@OUR@wrap#1\expandafter\endcsname
1734     \csname wrap#1\endcsname
1735     \caption@Error{%
1736         For a successful cooperation of the 'wrapfig' package\MessageBreak
1737         with the 'float' package you should load the 'wrapfig'\MessageBreak
1738         package *after* the 'float' package}%
1739     \else
1740     \expandafter\let\csname caption@ORI@wrap#1\expandafter\endcsname
1741     \csname wrap#1\endcsname
1742     \@namedef{wrap#1}{\caption@wrapfloat{#1}}%
1743     \expandafter\let\csname caption@OUR@wrap#1\expandafter\endcsname
1744     \csname wrap#1\endcsname
1745     \fi}%

```

```

\caption@wrapfloat
1746 \newcommand*\caption@wrapfloat[1]{%
1747     \caption@settype*{#1}%
1748     \float@ifstyle{#1}{%
1749         \ifx\WF@floatstyhook\undefined
1750         \caption@Error{%

```

```

1751         For a successful cooperation of the 'wrapfig' package\MessageBreak
1752         with the 'float' package you should use at least\MessageBreak
1753         'wrapfig' version 3.6}%
1754     \else
1755         \float@dostyle{#1}%
1756     \fi}}}%
1757 \caption@clearmargin
1758 %%% \caption@setoptions{wrapfloat}%
1759 \caption@setoptions{wrap#1}%
1760 \@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`.)

```

1761 \caption@restylewrapfloat{figure}%
1762 \caption@restylewrapfloat{table}%
1763 \caption@For{typelist}}{%
1764     \newenvironment{wrap#1}{\wrapfloat{#1}}{\endwrapfloat}%
1765     \caption@restylewrapfloat{#1}}%
1766 \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.

```

1767 \caption@ifundefined\float@restyle{}{%
1768     \toks@=\expandafter{\float@restyle{#1}% (env may or may not be defined)
1769     \caption@restylewrapfloat{#1}}%
1770     \edef\@tempa{\def\noexpand\float@restyle##1{\the\toks@}}%
1771     \@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`.

```

1772 \let\caption@ORI@wrapfloat\wrapfloat
1773 \def\wrapfloat#1{%
1774     \float@ifstyle{#1}{%
1775         \caption@Error{%
1776             For a successful cooperation of the 'wrapfig' package\MessageBreak
1777             with the 'float' package you should load the 'wrapfig'\MessageBreak
1778             package *right after* the 'float' package}}}%
1779     \caption@ORI@wrapfloat{#1}}%
1780 \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\bgrouper\setlength\hsize{#2}%
\ifdim\hsize>\z@ \@parboxrestore \else

```



```

\setbox\z@\hbox\bgroup \let\wf@@caption\caption \let\caption\wf@caption
\ignorespaces \fi}

```

Our code:

```

1781 \def\WF@rapt[#1]#2{% final two args: #1 = overhang, #2 = width,
1782 \gdef\WF@ovh{#1}% hold overhang for later, when \width is known
1783 \global\setbox\WF@box\vtop\bgroup \setlength\hsize{#2}%
1784 \caption@start
1785 \ifdim\hsize>\z@ \@parboxrestore \else
1786 \setbox\z@\hbox\bgroup \let\wf@@caption\caption \let\caption\wf@caption
1787 \ignorespaces \fi}%
1788 }{}

```

References

- [1] Frank Mittelbach and Michel Goossens:
The L^AT_EX 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^AT_EX 2_ε bundle,
2007-01-09
- [4] Victor Eijkhout:
An introduction to the Dutch L^AT_EX 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^AT_EX,
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 \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