

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 v3.2d, last revised 2011/11/02.

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	6
5.4	caption2 v2.x compatibility options	7
5.5	Obsolete caption v3.0 options	7
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	8
8	Processing of options	10
9	<code>\caption</code>, <code>\@caption</code>, and <code>\@makecaption</code>	10
10	<code>\captionof</code> and <code>\captionlistentry</code>	19
11	<code>\captionbox</code>	22
12	<code>\ContinuedFloat</code>	22
13	Internal helpers	24
14	Support for sub-captions	26
15	Document class & Babel package support	29
15.1	The $\mathcal{A}\mathcal{M}\mathcal{S}$ & SMF classes	29
15.2	The beamer class	29
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	31
16.1 The float package	33
16.2 The floatflt package	35
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	51
16.13 The supertabular and xtab packages	52
16.14 The threeparttable package	54
16.15 The wrapfig package	54

1 Identification

```
1 \NeedsTeXFormat{LaTeX2e}[1994/12/01]
2 \ProvidesPackage{caption}[2011/11/02 v3.2d Customizing captions (AR)]
3 %\@ifundefined{PackageRedefines}{}{\PackageRedefines{caption}{caption}}
```

2 Loading the kernel

```
4 \RequirePackage{caption3}[2011/08/30] % needs v1.4 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{%
41   \RequirePackage{newfloat}%
42   \newfloat@setname}
43 \newcommand*\caption@setname[2]{\@namedef{#1name}{#2}}
44 \caption@AtBeginDocument{\let\caption@SetName\caption@setname}
\caption@DeclareWithinOption
45 \newcommand*\caption@DeclareWithinOption[1]{%
46   \DeclareCaptionOption{#1within}{\caption@Within{#1}{##1}}%
47   \DeclareCaptionOptionNoValue{#1without}{\caption@Within{#1}{none}}}%
48 \@onlypreamble\caption@DeclareWithinOption
49 \caption@DeclareWithinOption{figure}
50 \caption@DeclareWithinOption{table}
51 \DeclareCaptionOption{within}{%
52   \RequirePackage{newfloat}%
53   \newfloatsetup{within=#1}}
54 \DeclareCaptionOptionNoValue{without}{%
55   \RequirePackage{newfloat}%
56   \newfloatsetup{without}}
\caption@Within
57 \newcommand*\caption@Within{%
58   \RequirePackage{newfloat}%
59   \newfloat@setwithin}

```

5.2 Miscellaneous options

```

60 \DeclareCaptionOption*{config}[caption]{%
61   \InputIfFileExists{#1.cfg}%
62   {\typeout{*** Local configuration file #1.cfg used ***}}%
63   {\caption@Warning{Configuration file #1.cfg not found}}}%
\caption@setparboxrestore \caption@setparboxrestore{<partial or full>}
64 \newcommand*\caption@setparboxrestore[1]{%
65   \caption@ifinlist{#1}{full}{%
66     \caption@setfullparboxrestore
67   }{\caption@ifinlist{#1}{default,light,partial}}%
68   \let\caption@parboxrestore\@secondoftwo
69 }{%
70   \caption@Error{Undefined parboxrestore `#1'}%
71 }}

```

```

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

74 \DeclareCaptionOption{parboxrestore}{\caption@setparboxrestore{#1}}
75 \captionsetup{parboxrestore=default}

76 \DeclareCaptionOption{@minipage}{%
77   \caption@ifinlist{#1}{auto,default}%
78   {\let\caption@if@minipage\@gobbletwo}%
79   {\caption@set@bool\caption@if@minipage{#1}}}
80 \captionsetup{@minipage=default}

```

5.3 caption v1.x compatibility options

```

81 \DeclareCaptionOption{compatibility}[1]{\caption@setbool{compatibility}{#1}}
82 \@onlypreamble@key{caption}{compatibility}

83 \DeclareCaptionOptionNoValue*{normal}{%
84   \caption@setformat{plain}%
85   \caption@setjustification{justified}}
86 \DeclareCaptionOptionNoValue*{isu}{%
87   \caption@setformat{hang}%
88   \caption@setjustification{justified}}
89 \DeclareCaptionOptionNoValue*{hang}{%
90   \caption@setformat{hang}%
91   \caption@setjustification{justified}}
92 \DeclareCaptionOptionNoValue*{center}{%
93   \caption@setformat{plain}%
94   \caption@setjustification{centering}}
95 \DeclareCaptionOptionNoValue*{anne}{%
96   \caption@setformat{plain}%
97   \caption@setjustification{centerlast}}
98 \DeclareCaptionOptionNoValue*{centerlast}{%
99   \caption@setformat{plain}%
100  \caption@setjustification{centerlast}}

101 \DeclareCaptionOptionNoValue*{scriptsize}{\def\captionfont{\scriptsize}}
102 \DeclareCaptionOptionNoValue*{footnotesize}{\def\captionfont{\footnotesize}}
103 \DeclareCaptionOptionNoValue*{small}{\def\captionfont{\small}}
104 \DeclareCaptionOptionNoValue*{normalsize}{\def\captionfont{\normalsize}}
105 \DeclareCaptionOptionNoValue*{large}{\def\captionfont{\large}}
106 \DeclareCaptionOptionNoValue*{Large}{\def\captionfont{\Large}}

107 \DeclareCaptionOptionNoValue*{up}{\l@addto@macro\captionlabelfont\upshape}
108 \DeclareCaptionOptionNoValue*{it}{\l@addto@macro\captionlabelfont\itshape}
109 \DeclareCaptionOptionNoValue*{sl}{\l@addto@macro\captionlabelfont\slshape}
110 \DeclareCaptionOptionNoValue*{sc}{\l@addto@macro\captionlabelfont\scshape}
111 \DeclareCaptionOptionNoValue*{md}{\l@addto@macro\captionlabelfont\mdseries}
112 \DeclareCaptionOptionNoValue*{bf}{\l@addto@macro\captionlabelfont\bfseries}
113 \DeclareCaptionOptionNoValue*{rm}{\l@addto@macro\captionlabelfont\rmfamily}
114 \DeclareCaptionOptionNoValue*{sf}{\l@addto@macro\captionlabelfont\sffamily}
115 \DeclareCaptionOptionNoValue*{tt}{\l@addto@macro\captionlabelfont\ttfamily}
116 \DeclareCaptionOptionNoValue*{nooneline}{\caption@setbool{slc}{0}}

```

```

117 \caption@setbool{ruled}{0}
118 \DeclareCaptionOptionNoValue*{ruled}{\caption@setbool{ruled}{1}}

```

5.4 caption2 v2.x compatibility options

```

119 \DeclareCaptionOptionNoValue*{flushleft}{%
120   \caption@setformat{plain}%
121   \caption@setjustification{raggedright}}
122 \DeclareCaptionOptionNoValue*{flushright}{%
123   \caption@setformat{plain}%
124   \caption@setjustification{raggedleft}}

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

126 \DeclareCaptionOptionNoValue*{ignoreLTcapwidth}{%
127   \caption@WarningNoLine{Obsolete option 'ignoreLTcapwidth' ignored}}

```

5.5 Obsolete caption v3.0 options

```

128 \DeclareCaptionOption*{caption}{%
129   \caption@setbool{temp}{#1}%
130   \caption@ifbool{temp}{}{
131     \caption@Error{%
132       The package option 'caption=#1' is obsolete.\MessageBreak
133       Please pass this option to the subfig package instead.\MessageBreak
134       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 `FPlist=caption` and `FPref=figure` which is inconsistent, but compatible to the usual behaviour of the `fltpage` package.

```

135 \DeclareCaptionOption{FPlist}[1]{\caption@setFPoption{list}{#1}}
136 \DeclareCaptionOption{FPref}[1]{\caption@setFPoption{ref}{#1}}
137 \@onlypreamble@key{caption}{FPlist}
138 \@onlypreamble@key{caption}{FPref}

139 \newcommand*\caption@setFPoption[2]{%
140   \edef\caption@tempa{\@car#2\@nil}%
141   \caption@setbool{FP#1cap}{\if c\caption@tempa 1\else 0\fi}}
142 \@onlypreamble\caption@setFPoption

143 \captionsetup{FPlist=caption,FPref=figure}

```

5.7 hyperref package support options

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

```

144 \DeclareCaptionOption{hypcap}[1]{\caption@setbool{hypcap}{#1}}
145 \DeclareCaptionOption{hypcapSPACE}{\def\caption@hypcapSPACE{#1}}

146 \captionsetup{hypcap=1,hypcapSPACE=.5\baselineskip}

```

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

```

147 \caption@ifamsclass{%
148   \caption@InfoNoLine{AMS or SMF document class}%

```

```

149 \setlength\belowcaptionskip{0pt}% set to 12pt by AMS class
150 }

```

7 KOMA-Script document classes support

```

151 \caption@ifkomaclass{%
152 \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
153 \g@addto@macro\@tablecaptionabovetrue{\captionsetup*[table]{position=t}}
154 \g@addto@macro\@tablecaptionabovefalse{\captionsetup*[table]{position=b}}

155 \if@tablecaptionabove
156 \@tablecaptionabovetrue
157 \else
158 \@tablecaptionabovefalse
159 \fi

\onelinecaptionstrue
\onelinecaptionfalse
160 \g@addto@macro\onelinecaptionstrue{\let\caption@ifslc\@firstoftwo}
161 \g@addto@macro\onelinecaptionfalse{\let\caption@ifslc\@secondoftwo}

162 \ifonelinecaptions
163 \onelinecaptionstrue
164 \else
165 \onelinecaptionfalse
166 \fi

\@captionabovetrue
\@captionabovefalse
Please note that these are stronger than the position setting, therefore we override the
options figureposition and tableposition to typeout a warning.
167 \g@addto@macro\@captionabovetrue{\let\caption@position\@firstoftwo}
168 \g@addto@macro\@captionabovefalse{\let\caption@position\@secondoftwo}

169 \DeclareCaptionOption{figureposition}{%
170 \caption@WarningNoLine{Option 'figureposition=#1' has no effect\MessageBreak
171 when used with a KOMA script document class}}
172 \DeclareCaptionOption{tableposition}{%
173 \caption@WarningNoLine{Option 'tableposition=#1' has no effect\MessageBreak
174 when used with a KOMA script document class}}

\setcapindent
175 \let\caption@KOMA@setcapindent\@setcapindent
176 \renewcommand*\@setcapindent[1]{%
177 \caption@KOMA@setcapindent{#1}\caption@setcapindent}

178 \let\caption@KOMA@@setcapindent\@@setcapindent
179 \renewcommand*\@@setcapindent[1]{%
180 \caption@KOMA@@setcapindent{#1}\caption@setcapindent}

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

183 \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.)

```

184 \expandafter\let\expandafter\caption@KOMA@setcapwidth
185       \csname\string\setcapwidth\endcsname
186 \@namedef{\string\setcapwidth}[#1]#2{%
187   \caption@KOMA@setcapwidth[#1]{#2}\caption@setcapwidth{#1}}
188 \newcommand*\caption@setcapwidth[1]{%
189   \ifx\\#1\\\else
190     \caption@ifundefined\cap@margin{%
191       \def\@tempa{captionbeside}%
192       \ifx\@tempa\@currentenv\else\caption@Warning{%
193         Ignoring optional argument [#1] of \string\setcapwidth\MessageBreak}%
194       \fi}%
195   \fi
196   \captionsetup{width=\cap@width}}

197 \def\caption@tempa{\hsize}%
198 \ifx\caption@tempa\cap@width \else
199   \caption@setcapwidth{?}
200 \fi

```

`\setcapmargin`

```

201 \expandafter\let\expandafter\caption@KOMA@setcapmargin
202       \csname\string\@setcapmargin\endcsname
203 \@namedef{\string\@setcapmargin}[#1]#2{%
204   \caption@KOMA@setcapmargin[#1]{#2}\caption@setcapmargin}

205 \expandafter\let\expandafter\caption@KOMA@@setcapmargin
206       \csname\string\@@setcapmargin\endcsname
207 \@namedef{\string\@@setcapmargin}[#1]#2{%
208   \caption@KOMA@@setcapmargin[#1]{#2}\caption@setcapmargin}

209 \newcommand*\caption@setcapmargin{%
210   \begingroup
211     \let\onelinecaptionsfalse\relax
212     \def\@twoside{0}%
213     \def\if@twoside{\def\@twoside{1}\iffalse}%
214     \cap@margin
215     \def\@tempa{\endgroup}%
216     \ifx\cap@left\hfill\else\ifx\cap@right\hfill\else
217       \def\hspace##1##{\@firstofone}%
218       \edef\@tempa{\endgroup
219         \noexpand\captionsetup{%
220           twoside=\@twoside,slc=0,%
221           margin={\cap@left,\cap@right}}}%
222     \fi\fi
223     \@tempa}

224 \ifx\cap@margin\relax \else
225   \caption@setcapmargin
226 \fi

227 }

```

8 Processing of options

```
228 \caption@SetupOptions{caption}{\caption@setkeys{#1}{#2}}
229 \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`.

```
230 \def\caption@caption{%
231   \caption@iftype
232     {\caption@checkgrouplevel\@empty\caption
233     \caption@star
234       {\@nameuse{donemaincaptiontrue}%
235       \caption@refstepcounter\@capttype}%
236     {\caption@dblarg{\@caption\@capttype}}}%
237   {\caption@Error{\noexpand\caption outside float}%
238   \caption@gobble}}
```

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

```
239 \newcommand*\caption@star[2]{%
240   \@ifstar{\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](#).)

```
241 \long\def\caption@@caption#1[#2]#3{%
242   \ifcaption@star \else
243     \caption@prepareanchor{#1}{#2}%
244     \M@getttitle{#2}%
245     \memcaptioninfo{#1}{\csname the#1\endcsname}{#2}{#3}%
246   \fi
247   \par
248   \caption@beginex{#1}{#2}{#3}%
249   \caption@setfloatcapt{%
250     \caption@boxrestore
251     \if@minipage
252       \@setminipage
253     \fi
254     \caption@normalsize
255     \ifcaption@star
256       \let\caption@makeanchor\@firstofone
257     \fi
258     \@makecaption{\csname fnum#1\endcsname}%
259     {\ignorespaces\caption@makeanchor{#3}}\par
260     \caption@if@minipage\@minipagetrue\@minipagefalse}%
261   \caption@end}
```

memoir document class stuff:

```
262 \providecommand\M@getttitle[1]{}
263 \providecommand\memcaptioninfo[4]{}

```

\caption@prepareanchor

```
264 \newcommand*\caption@prepareanchor[2]{%
265   \caption@makecurrent{#1}{#2}%
266   \caption@ifhyppcap\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.

```
267 \long\def\caption@makecaption#1#2{%
268   \caption@iftop
269     {\vskip\belowcaptionskip}%
270     {\caption@rule\vskip\abovecaptionskip}%
271   \caption@@make{#1}{#2}%
272   \caption@iftop
273     {\vskip\abovecaptionskip\caption@rule}%
274     {\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.

```
275 \newcommand*\caption@redefine{}
276 \g@addto@macro\caption@redefine{%
277   \caption@setbool{incompatible}{0}%
278   \caption@CheckCommand\caption{%
279     % ltfloat.dtx [2002/10/01 v1.1v LaTeX Kernel (Floats)]
280     \def\caption{%
281       \ifx\@captive\@undefined
282         \@latex@error{\noexpand\caption outside float}\@ehd
283         \expandafter\@gobble
284       \else
285         \refstepcounter\@captive
286         \expandafter\@firstofone
287       \fi
288       {\@dblarg{\@caption\@captive}}%
289     }%
290   \caption@CheckCommand\caption{%
291     % beamerbaselocalstructure.sty,v 1.53 2007/01/28 20:48:21 tantau
292     \def\caption{
293       \ifx\@captive\@undefined
294         \@latex@error{\noexpand\caption outside figure or table}\@ehd
295         \expandafter\@gobble
296       \else
297         \refstepcounter\@captive
298         \expandafter\@firstofone
299       \fi

```

```

300     {\@dblarg{\@caption\@capttype}}%
301   }}%

302 \caption@CheckCommand\caption{%
303   % float.sty [2001/11/08 v1.3d Float enhancements (AL)]
304   \renewcommand\caption{%
305     \ifx\@capttype\@undefined
306       \@latex@error{\noexpand\caption outside float}\@ehd
307       \expandafter\@gobble
308     \else
309       \refstepcounter\@capttype
310       \let\@tempf\@caption
311       \expandafter\ifx\csname @float@c@\@capttype\endcsname\relax\else
312         \expandafter\expandafter\let
313         \expandafter\@tempf\csname @float@c@\@capttype\endcsname
314       \fi
315     \fi
316     \@dblarg{\@tempf\@capttype}}}%

317 \caption@CheckCommand\caption{%
318   % hyperref.sty [2007/02/27 v6.75t Hypertext links for LaTeX]
319   % hyperref.sty [2007/04/09 v6.76a Hypertext links for LaTeX]
320   % hyperref.sty [2007/06/12 v6.76h Hypertext links for LaTeX]
321   \def\caption{%
322     \ifx\@capttype\@undefined
323       \@latex@error{\noexpand\caption outside float}\@ehd
324       \expandafter\@gobble
325     \else
326       \H@refstepcounter\@capttype
327       \@ifundefined{fst@\@capttype}{%
328         \let\Hy@tempa\@caption
329       }{%
330         \let\Hy@tempa\Hy@float@caption
331       }%
332       \expandafter\@firstofone
333     \fi
334     {\@dblarg{\Hy@tempa\@capttype}}%
335   }}%

336 \caption@CheckCommand\caption{%
337   % hyperref.sty [2007/08/05 v6.76j Hypertext links for LaTeX]
338   \def\caption{%
339     \ifx\@capttype\@undefined
340       \@latex@error{\noexpand\caption outside float}\@ehd
341       \expandafter\@gobble
342     \else
343       \H@refstepcounter\@capttype
344       \let\Hy@tempa\@caption
345       \@ifundefined{float@caption}{%
346       }{%
347         \expandafter\ifx\csname @float@c@\@capttype\endcsname\float@caption
348         \let\Hy@tempa\Hy@float@caption
349       \fi
350       }%
351       \expandafter\@firstofone
352     \fi

```

```

353     {\@dblarg{\Hy@tempa\@capttype}}%
354   }}%

355   \caption@CheckCommand\caption{%
356     % memhfixc.sty [2010/08/17 v1.15 nameref/hyperref package fixes for memoir cl
357     % \let\m@moldhycaption\caption
358     \renewcommand{\caption}{\donemaincaptiontrue\m@moldhycaption}}%

359   \caption@IfCheckCommand{}{%
360     \caption@InfoNoLine{%
361       Incompatible package detected (regarding \string\caption).\MessageBreak
362       \string\caption\space=\space\meaning\caption}%
363     \caption@setbool{incompatible}{1}}%

364   \caption@CheckCommand\@caption{%
365     % ltfloating.dtx [2002/10/01 v1.1v LaTeX Kernel (Floats)]
366     \long\def\@caption#1[#2]#3{%
367       \par
368       \addcontentsline{\csname ext@#1\endcsname}{#1}%
369       {\protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}}%
370       \begingroup
371       \@parboxrestore
372       \if@minipage
373         \@setminipage
374       \fi
375       \normalsize
376       \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
377     \endgroup}}%

378   \caption@CheckCommand\@caption{%
379     % beamerbaselocalstructure.sty,v 1.53 2007/01/28 20:48:21 tantau
380     \long\def\@caption#1[#2]#3{% second argument ignored
381       \par\nobreak
382       \begingroup
383       \@parboxrestore
384       \if@minipage
385         \@setminipage
386       \fi
387       \beamer@makecaption{#1}{\ignorespaces #3}\par\nobreak
388     \endgroup}}%

389   \caption@CheckCommand\@caption{%
390     % magyar.ldf [2005/03/30 v1.4j Magyar support from the babel system]
391     \long\def\@caption#1[#2]#3{%
392       \csname par\endcsname
393       \addcontentsline{\csname ext@#1\endcsname}{#1}%
394       {\protect\numberline{\csname the#1\endcsname.}{\ignorespaces #2}}%
395       \begingroup
396       \@parboxrestore
397       \if@minipage
398         \@setminipage
399       \fi
400       \normalsize
401       \@makecaption{\csname fnum@#1\endcsname}%
402       {\ignorespaces #3}\csname par\endcsname
403     \endgroup}}%

404 %   \caption@CheckCommand\float@caption{%

```

```

405 % % float.sty [2001/11/08 v1.3d Float enhancements (AL)]
406 % \long\def\float@caption#1[#2]#3{%
407 %     \addcontentsline{\@nameuse{ext@#1}}{#1}%
408 %     {\protect\numberline{\@nameuse{the#1}}{\ignorespaces #2}}
409 %     \global\setbox\@floatcapt\vbox\bgroup\@parboxrestore
410 %     \normalsize\@fs@capt{\@nameuse{fnum@#1}}{\ignorespaces #3}%
411 %     \ifnextchar[{\float@ccon}{\egroup}}%
412 %     \long\def\float@ccon[#1]{#1\par\egroup}}%

413 \caption@CheckCommand\@caption{%
414 % hyperref.sty [2007/02/27 v6.75t Hypertext links for LaTeX]
415 \long\def\@caption#1[#2]#3{%
416     \hyper@makecurrent{\@capttype}%
417     \def\@currentlabelname{#2}%
418     \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
419         \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
420     }%
421     \begingroup
422         \@parboxrestore
423         \if@minipage
424             \setminipage
425         \fi
426         \normalsize
427         \makecaption{\csname fnum@#1\endcsname}{%
428             \ignorespaces
429             \ifHy@nesting
430                 \hyper@@anchor{\@currentHref}{#3}%
431             \else
432                 \Hy@raisedlink{\hyper@@anchor{\@currentHref}{\relax}}#3%
433             \fi
434         }%
435     \par
436 \endgroup
437 }}%

438 \caption@CheckCommand\@caption{%
439 % hyperref.sty [2007/04/09 v6.76a Hypertext links for LaTeX]
440 % hyperref.sty [2007/06/12 v6.76h Hypertext links for LaTeX]
441 % hyperref.sty [2007/08/05 v6.76j Hypertext links for LaTeX]
442 \long\def\@caption#1[#2]#3{%
443     \expandafter\ifx\csname if@capstart\expandafter\endcsname
444         \csname iftrue\endcsname
445     \global\let\@currentHref\hc@currentHref
446 \else
447     \hyper@makecurrent{\@capttype}%
448 \fi
449 \def\@currentlabelname{#2}%
450 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
451     \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
452 }%
453 \begingroup
454     \@parboxrestore
455     \if@minipage
456         \setminipage
457     \fi

```

```

458         \normalsize
459         \expandafter\ifx\csname if@capstart\expandafter\endcsname
460             \csname iftrue\endcsname
461             \global\@capstartfalse
462             \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
463         \else
464             \@makecaption{\csname fnum@#1\endcsname}{%
465                 \ignorespaces
466                 \ifHy@nesting
467                     \hyper@@anchor{\@currentHref}{#3}%
468                 \else
469                     \Hy@raisedlink{\hyper@@anchor{\@currentHref}{\relax}}#3%
470                 \fi
471             }%
472         \fi
473     \par
474 \endgroup
475 }%

476 \caption@CheckCommand\@caption{%
477     % hyperref.sty [2009/11/27 v6.79k Hypertext links for LaTeX]
478     \long\def\@caption#1[#2]#3{%
479         \expandafter\ifx\csname if@capstart\expandafter\endcsname
480             \csname iftrue\endcsname
481             \global\let\@currentHref\hc@currentHref
482         \else
483             \hyper@makecurrent{\@capttype}%
484         \fi
485         \def\@currentlabelname{#2}%
486         \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
487             \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
488         }%
489         \begingroup
490             \@parboxrestore
491             \if@minipage
492                 \@setminipage
493             \fi
494             \normalsize
495             \expandafter\ifx\csname if@capstart\expandafter\endcsname
496                 \csname iftrue\endcsname
497                 \global\@capstartfalse
498                 \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
499             \else
500                 \@makecaption{\csname fnum@#1\endcsname}{%
501                     \ignorespaces
502                     \ifHy@nesting
503                         \expandafter\hyper@@anchor\expandafter{\@currentHref}{#3}%
504                     \else
505                         \Hy@raisedlink{%
506                             \expandafter\hyper@@anchor\expandafter{\@currentHref}{\relax}%
507                         }%
508                     #3%
509                 \fi
510             }%
511         \fi

```

```

512     \par
513     \endgroup
514 } }%

515 \caption@CheckCommand\@caption{%
516 % hyperref.sty [2009/12/09 v6.79m Hypertext links for LaTeX]
517 % hyperref.sty [2009/12/28 v6.79z Hypertext links for LaTeX]
518 \long\def\@caption#1[#2]#3{%
519     \expandafter\ifx\csname if@capstart\expandafter\endcsname
520         \csname iftrue\endcsname
521     \global\let\@currentHref\hc@currentHref
522 \else
523     \hyper@makecurrent{\@captype}%
524 \fi
525 \@ifundefined{NR@getttitle}{%
526     \def\@currentlabelname{#2}%
527 }{%
528     \NR@getttitle{#2}%
529 }%
530 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
531     \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
532 }%
533 \begingroup
534     \@parboxrestore
535     \if@minipage
536         \@setminipage
537     \fi
538     \normalsize
539     \expandafter\ifx\csname if@capstart\expandafter\endcsname
540         \csname iftrue\endcsname
541     \global\@capstartfalse
542     \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
543 \else
544     \@makecaption{\csname fnum@#1\endcsname}{%
545         \ignorespaces
546         \ifHy@nesting
547             \expandafter\hyper@@anchor\expandafter{\@currentHref}{#3}%
548         \else
549             \Hy@raisedlink{%
550                 \expandafter\hyper@@anchor\expandafter{%
551                     \@currentHref
552                 }{\relax}%
553             }%
554             #3%
555         \fi
556     }%
557 \fi
558 \par
559 \endgroup
560 } }%

561 \caption@CheckCommand\@caption{%
562 % nameref.sty [2006/12/27 v2.28 Cross-referencing by name of section]
563 \long\def\@caption#1[#2]{%
564     \def\@currentlabelname{#2}%

```



```

565     \NR@@caption{#1} [{#2}]%
566   }}%

567 \caption@CheckCommand\@caption{%
568   % nameref.sty [2009/11/27 v2.32 Cross-referencing by name of section]
569   \long\def\@caption#1[#2]{%
570     \NR@getttitle{#2}%
571     \NR@@caption{#1} [{#2}]%
572   }}%

573 \caption@CheckCommand\@caption{%
574   % subfigure.sty [2002/07/30 v2.1.4 subfigure package]
575   \long\def\@caption#1[#2]#3{%
576     \ifundefined{if#1topcap}%
577       {\subfig@oldcaption{#1} [{#2}] {#3}}%
578       {\@nameuse{if#1topcap}%
579         \@listsubcaptions{#1}%
580         \subfig@oldcaption{#1} [{#2}] {#3}%
581         \else
582         \subfig@oldcaption{#1} [{#2}] {#3}%
583         \@listsubcaptions{#1}%
584         \fi}}}%

585 \caption@CheckCommand\@caption{%
586   % subfig.sty [2005/06/28 ver: 1.3 subfig package]
587   \def\@caption{\caption@}%
588 %   \long\def\caption@#1[#2]#3{%
589 %     \ifundefined{caption@setfloattype}%
590 %       \caption@settype
591 %       \caption@setfloattype
592 %       \@capytype
593 %     \sf@ifpositiontop{%
594 %       \@listsubcaptions{#1}%
595 %       \sf@old@caption{#1} [{#2}] {#3}%
596 %     }{%
597 %       \sf@old@caption{#1} [{#2}] {#3}%
598 %       \@listsubcaptions{#1}%
599 %     }}%
600   }%

601 \caption@IfCheckCommand{}{%
602   \caption@InfoNoLine{%
603     Incompatible package detected (regarding \string\@caption).\MessageBreak
604     \string\@caption\space=\space\meaning\@caption}%
605   \caption@setbool{incompatible}{1}}%

```

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

```

606 \caption@ifundefined\caption@ifcompatibility
607   {\let\caption@ifcompatibility\caption@ifincompatible
608     \let\caption@tempa\caption@WarningNoLine}%
609   {\let\caption@tempa\@gobble}% suppress warning
610 \caption@ifcompatibility{%
611   \caption@tempa{%
612     \noexpand\caption will not be redefined since it's already\MessageBreak
613     redefined by a document class or package which is\MessageBreak
614     unknown to the caption package}%

```

```

615 \renewcommand*\caption@redefine{%
\ContinuedFloat is not supported in compatibility mode.
616 \renewcommand*\caption@ContinuedFloat[1]{%
617 \caption@Error{Not available in compatibility mode}}%
\caption@start is not supported in compatibility mode.
618 \caption@AtBeginDocument*{%
619 \let\caption@start\relax
620 \caption@ifundefined\caption@ORI@capstart{}{%
621 \caption@Debug{%
622 Restore hypcap definition of \string\capstart\@gobble}%
623 \let\capstart\caption@ORI@capstart}%
624 \caption@ifundefined\caption@ORI@float@makebox{}{%
625 \caption@Debug{%
626 Restore hyperref redefinition of \string\float@makebox\@gobble}%
627 \let\float@makebox\caption@ORI@float@makebox}%
628 }%

\caption@star We redefine \caption@star here so it does not make any harm.
629 \renewcommand*\caption@star[2]{#1#2}%

630 }{%
631 \caption@ifincompatible{%
632 \caption@WarningNoLine{%
633 Forced redefinition of \noexpand\caption since the\MessageBreak
634 unsupported(!) package option 'compatibility=false'\MessageBreak
635 was given}%
636 }{}%

\caption
\@caption 637 \renewcommand*\caption@redefine{%
638 \let\caption\caption@caption
639 \let\@caption\caption@@caption}%
640 \caption@redefine

641 }%
642 \caption@AtBeginDocument*{%
643 \let\caption@ORI@capstart\@undefined
644 \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.
645 \let\caption@ORI@xfloat\@xfloat
646 \def\@xfloat#1[#2]{%
647 \caption@ORI@xfloat{#1}[#2]%
648 \caption@settype{#1}}%

649 }

```

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.

```

650 \caption@AtBeginDocument{\caption@redefine}

```

\@makecaption

```
651 \let \@makecaption \caption@makecaption
```

10 \captionof and \captionlistentry

```
652 \caption@AtBeginDocument{%
653   \DeclareCaptionOption{type}{\setcaptiontype{#1}}%
654   \DeclareCaptionOption{type*}{\setcaptiontype*{#1}}%
655   \DeclareCaptionOptionNoValue{subtype}{\setcaptionsubtype\relax}%
656   \DeclareCaptionOptionNoValue{subtype*}{\setcaptionsubtype*}%
657 }
```

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.

```
658 \newcommand\setcaptiontype{%
659   \caption@boxrestore@mini
660   \caption@settype}
```

\setcaptionsubtype Same, but sets the sub-type.

```
661 \newcommand\setcaptionsubtype{%
662   \caption@iftype
663   \caption@setsubtype
664   {\caption@Error{\noexpand\setcaptionsubtype outside float}}}%
665 \newcommand\caption@setsubtype{%
666   \@ifstar
667   {\caption@@settype{sub}*{sub\@capttype}}%
668   {\caption@@settype{sub}{sub\@capttype}}}%

```

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

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

```
669 \newcommand*\caption@settype{%
670   \caption@@settype{}}
671 \newcommand*\caption@@settype[1]{%
672   \caption@teststar{\caption@@@settype{#1}}{\@firstoftwo\@secondoftwo}
673 \newcommand*\caption@@@settype[3]{%
674   % #1 = "" or "sub"
675   % #2 = \@firstoftwo in star form, \@secondoftwo otherwise
676   % #3 = <type>, e.g. "figure" or "table"
677   \caption@Debug{#1type=#3}%
678   \caption@checkgrouplevel{#1}{%
679     \captionsetup{#1type#2*\@empty=...}#2{ or
680       \@backslashchar#1captionof}}}%

```

```

681 \edef\caption@tempa{#3}%
682 \expandafter\ifx\csname @#1capttype\endcsname\caption@tempa \else
683   \ifcaptionsetup@star\else\@nameuse{caption@#1type@warning}\fi
684 \fi
685 \expandafter\let\csname @#1capttype\endcsname\caption@tempa
686 \@nameuse{caption@#1typehook}%
687 \caption@setoptions{#3}%
688 \ifx\caption@opt\relax
689   \@nameundef{caption@#1type@warning}%
690 \else
691   \@namedef{caption@#1type@warning}{\caption@Warning{%
692     The #1caption type was already set to
693     '\csname @#1capttype\endcsname'\MessageBreak}}%
694 \fi

695 \let\caption@ifrefstepcounter\@secondoftwo
696 #2{}{%
697   \let\@currentlabel\caption@undefinedlabel
698 % \let\@currentHlabel\@undefined
699   \ifx\caption@x@label\@undefined
700     \let\caption@x@label\label
701     \let\label\caption@xlabel
702   \fi
703   \caption@start}}

```

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

```
704 \newcommand*\caption@typehook{}
```

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

```

705 \newcommand*\caption@iftype{%
706   \caption@ifundefined\@capttype\@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)

```

707 \begingroup\expandafter\expandafter\expandafter\endgroup
708 \expandafter\ifx\csname currentgrouplevel\endcsname\relax
709   \caption@Debug{TeX engine: TeX}
710   \let\caption@checkgrouplevel\@gobbletwo
711 \else
712   \caption@Debug{TeX engine: e-TeX}
713   \newcommand*\caption@checkgrouplevel[2]{%
714     \ifundefined{#1caption@grouplevel}{%
715       \caption@ifundefined\caption@grouplevel{\let\caption@grouplevel\z@}{}%
716       \ifnum\currentgrouplevel>\caption@grouplevel\relax
717         \expandafter\edef\csname #1caption@grouplevel\endcsname{%
718           \the\currentgrouplevel}%
719       \else
720         \caption@Warning{\string#2\MessageBreak outside box or environment}%
721       \fi
722     }{}%
723 \fi

```

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

```

724 \newcommand*\caption@undefinedlabel{%
725   \protect\caption@xref{\caption@labelname}{\on@line}}

726 \DeclareRobustCommand*\caption@xref[2]{%
727   \caption@WarningNoLine{\noexpand\label without proper \string\caption#2}%
728   \@setref\relax\@undefined{#1}}

729 \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.

```

730 \newcommand*\caption@xlabel{%
731   \caption@withoptargs\caption@@xlabel}

732 \newcommand*\caption@@xlabel[2]{%
733   \caption@@@xlabel
734   \def\caption@labelname{#2}%
735   \caption@x@label#1{#2}}

736 \newcommand*\caption@@@xlabel{%
737   \global\let\caption@@@xlabel\@empty
738   \@bsphack
739   \protected@write\@auxout{}%
740     {\string\providecommand*\string\caption@xref[2]{%
741       \string\@setref\string\relax\string\@undefined{\string#1}}}%
742   \@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)

```

743 \caption@AtBeginDocument{%
744   \def\captionof{\caption@teststar\caption@of{\caption*}\caption}}

745 \newcommand*\caption@of[2]{\setcaptiontype*{#2}#1}

```

`\captionlistentry` `\captionlistentry[<float type>]{<list entry>}`
`\captionlistentry* [<float type>]{<list entry>}`

```

746 \newcommand*\captionlistentry{%
747   \caption@teststar\@captionlistentry\@firstoftwo\@secondoftwo}

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

750 \def\caption@listentry#1[#2]#3{%
751   \@bsphack
752   #1{\caption@getttitle{#3}}%
753   {\caption@refstepcounter{#2}}%
754   \caption@makecurrent{#2}{#3}}%
755   \caption@addcontentsline{#2}{#3}%
756   \@esphack}

```

11 \captionbox

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

```

757 \newcommand*\captionbox{%
758   \let\captionbox@settype\@gobble
759   \caption@withoptargs\caption@box}

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

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

764 \long\def\caption@iibox#1#2#3[#4]#5{%
765   \setbox\@tempboxa\hbox{#5}%
766   \begingroup
767     \captionbox@settype*% set \caption@position
768     \caption@iftop{%
769       \endgroup
770       \parbox[t]{#3}{%
771         \captionbox@settype\relax
772         \caption@setposition t%
773         \vbox{\caption#1{#2}}%
774         \captionbox@hrule
775         \csname caption@hj@#4\endcsname
776         \unhbox\@tempboxa}%
777       }{%
778         \endgroup
779         \parbox[b]{#3}{%
780           \captionbox@settype\relax
781           \caption@setposition b%
782           \csname caption@hj@#4\endcsname
783           \unhbox\@tempboxa
784           \captionbox@hrule
785           \vtop{\caption#1{#2}}}%
786     }}

787 \newcommand*\captionbox@hj@default{c}
788 \newcommand*\captionbox@hrule{\hrule\@height\z@}

789 \providecommand*\caption@hj@c{\centering}
790 \providecommand*\caption@hj@l{\raggedright}
791 \providecommand*\caption@hj@r{\raggedleft}
792 \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

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

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

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

```

793 \def\ContinuedFloat{%
794   \@ifnextchar[\@Continued@Float\@ContinuedFloat}
795 \def\@Continued@Float[#1]{\addtocounter{#1}{m@ne}
796 \def\@ContinuedFloat{%
797   \caption@iftype
798     {\addtocounter{\@c@pttype}{m@ne}
799     \caption@ContinuedFloat{\@c@pttype}%
800     {\caption@Error{\noexpand\ContinuedFloat outside float}}}}
801 \def\caption@ContinuedFloat#1{%
802   \@ifstar{\caption@Continued@Float@{#1}}{\caption@Continued@Float{#1}}}
803 \def\caption@Continued@Float@{%
804   \addtocounter{\@c@pttype}{@ne}
805   \@stpelt{ContinuedFloat}\stepcounter{ContinuedFloat}%
806   \def\caption@resetContinuedFloat##1{\xdef\caption@CFtype{##1}}%
807   \caption@@ContinuedFloat}
808 \def\caption@Continued@Float#1{%
809   \edef\caption@tempa{#1}%
810   \ifx\caption@tempa\caption@CFtype
811     \stepcounter{ContinuedFloat}%
812     \let\caption@resetContinuedFloat\@gobble
813     \caption@@ContinuedFloat{#1}%
814     \sf@ContinuedFloat{#1}%
815   \else
816     \caption@Error{Continued ‘#1’ after ‘\caption@CFtype’}%
817   \fi}
818 \def\caption@@ContinuedFloat#1{%
819   \expandafter\l@addto@macro\c@name the#1\endcsname\theContinuedFloat
820   \@ifundefined{theH#1}{}{%
821     \expandafter\l@addto@macro\c@name theH#1\endcsname{%
822       \@alpha@c@ContinuedFloat}}%
823   \caption@setoptions{ContinuedFloat}%
824   \caption@setoptions{continued#1}}
825 \providecommand*\sf@ContinuedFloat[1]{}
826 \newcommand*\caption@CFtype{??}

```

<code>\theContinuedFloat</code>	<p>Its preset to <code>\@empty</code>, so usually the continuation counter is not included in the caption label or references.</p> <pre>827 \newcounter{ContinuedFloat} 828 \let\theContinuedFloat\@empty</pre>
<code>\caption@resetContinuedFloat</code>	<pre>\caption@resetContinuedFloat{<type>} If a continuation counter is defined, we reset it. (This one will be called inside \@caption.) 829 \newcommand*\caption@resetContinuedFloat[1]{% 830 \@stpelt{ContinuedFloat}\xdef\caption@CFtype{#1}}</pre>
<code>\phantomcaption</code>	<pre>\phantomcaption Use this one for figures with subcaptions but without main caption. 831 \newcommand\phantomcaption{% 832 \caption@iftype 833 {\caption@refstepcounter\@captype}% 834 {\caption@Error{\noexpand\phantomcaption outside float}}}%</pre>

13 Internal helpers

<code>\caption@refstepcounter</code>	<p>Resets the continuation counter, increments the float (i.e. figure or table) counter, and sets the <code>refstepcounter</code> flag.</p> <pre>835 \newcommand*\caption@refstepcounter[1]{% 836 \@ifundefined{c@#1}% 837 {\caption@Error{No float type ‘#1’ defined}}% 838 {\caption@resetContinuedFloat{#1}% 839 \caption@@refstepcounter{#1}% 840 \let\caption@ifrefstepcounter\@firstoftwo}} 841 \newcommand*\caption@@refstepcounter{\refstepcounter} 842 \let\caption@ifrefstepcounter\@secondoftwo</pre>
<code>\caption@dblarg</code>	<p>A <code>\relax</code> was added compared to <code>\@dblarg</code> so <code>\caption{}</code> will be expanded to <code>\caption[\relax]{}</code> (and not to <code>\caption[]{}).</code></p> <pre>843 \caption@ifundefined\kernel@ifnextchar 844 {\newcommand\caption@dblarg[1]{\@ifnextchar[{\#1}{\caption@xdblarg{#1}}}% 845 {\newcommand\caption@dblarg[1]{\kernel@ifnextchar[{\#1}{\caption@xdblarg{#1}}}}}% 846 \newcommand\caption@xdblarg[2]{\#1[{\#2\relax}]{\#2}}%</pre>
<code>\caption@begin</code>	<p>Our handling of <code>\caption</code> will always be surrounded by <code>\caption@begin</code> (or <code>\caption@beginex</code>) and <code>\caption@end</code>. <code>\caption@begin{<type>}</code> performs these tasks:</p> <ol style="list-style-type: none"> 1. Start a new group. 2. Define <code>\fnum{<type>}</code> if the caption label format is set to non-default. 3. Override the <code>position=</code> setting, if necessary. (for example if set to <code>auto</code> or used inside a <code>supertabular</code>) <pre>847 \newcommand*\caption@begin[1]{% 848 \begingroup 849 \caption@setfnum{#1}% 850 \caption@fixposition 851 \global\let\caption@fixedposition\caption@position}</pre>

`\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>`.

```
852 \newcommand\caption@beginex[3]{%
853   \caption@begin{#1}%
854   \let\lst@@caption\relax
855   \caption@addcontentsline{#1}{#2}%
856   \caption@ifempty{#3}{}}
```

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

```
857 \newcommand*\caption@end{%
858   \endgroup
859   \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.

```
860 \newcommand*\caption@setfnum[1]{%
861   \@ifundefined{fnum@#1}{\iftrue}{\ifx\caption@lfmt\caption@lfmt@default\else}%
862   \@namedef{fnum@#1}{\caption@fnum{#1}}%
863   \fi}
```

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

```
\def\@parboxrestore{\@arrayparboxrestore\let\\\@normalcr}
\def\@arrayparboxrestore{%
  \let@if@nbreak\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`.

```
864 \newcommand*\caption@boxrestore{%
865   \caption@parboxrestore{\@parboxrestore}{%
866     \let@if@nbreak\iffalse
867     \let@if@noskipsec\iffalse
868     \let\par\@par
869 %   \let\-\@dischyph
870 %   \let'\@acci\let'\@accii\let\=\@acciii}
```

```

871 \parindent\z@ \parskip\z@skip
872 \everypar{}%
873 % \linewidth\hsize
874 % \@totalleftmargin\z@
875 \leftskip\z@skip \rightskip\z@skip \@rightskip\z@skip
876 \parfillskip\@flushglue \lineskip\normallineskip
877 \baselineskip\normalbaselineskip
878 \sloppy
879 \let\\\@normalcr
880 }}

```

`\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`.

```

881 \newcommand\caption@boxrestore@mini{%
882 \let\par\@par
883 \parindent\z@ \parskip\z@skip
884 \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).

```

885 \newcommand*\caption@normalsize{%
886 \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.

```
887 \let\caption@setfloatcapt\@firstofone
```

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

```

\caption@makeanchor 888 \newcommand*\caption@makecurrent[2]{}
\caption@start      889 \let\caption@makeanchor\@firstofone
\caption@@start     890 \let\caption@start\relax
\caption@freezeHref 891 \let\caption@@start\relax
\caption@defrostHref 892 \let\caption@freezeHref\relax
\caption@defrostHref 893 \let\caption@defrostHref\relax

```

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

```

894 \newcommand\caption@getttitle[1]{%
895 \caption@ifundefined\NR@getttitle
896 {\def\@currentlabelname{#1}}%
897 {\NR@getttitle{#1}}}

```

14 Support for sub-captions

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

```

898 \def\caption@DeclareSubType sub#1\@nil{%
899 \caption@Debug{Initializing subtype for `#1'\@gobble}%
900 \@namedef{caption@c@#1}{0}%
901 \@namedef{caption@beginsub#1}{\caption@beginsubfloat{#1}}
902 \@onlypreamble\caption@DeclareSubType

```

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

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

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

```
904 \caption@AtBeginDocument*{%
905   \caption@ifundefined\sf@counterlist{}{%
906     \@for\sf@temp:=\sf@counterlist\do{%
907       \expandafter\caption@DeclareSubType\sf@temp\@nil}}}
```

\caption@subtypehook

Hook, will be used inside \caption@setsubtype.

(Note: If we are inside an subfloatrow environment we have to keep the \@makecaption code of the floatrow package intact.)

```
908 \newcommand*\caption@subtypehook{%
909   \ifx\caption\caption@subcaption \else
910     \caption@warmup
911     \caption@ifrefstepcounter{}{%
912       % no \caption or \subcaption in this (floating) environment yet
913       \caption@Debug{Increment \@c@type\ counter =\the\value\@c@type}%
914       \caption@l@stepcounter\@c@type
915       \let\caption@@@addcontentsline\caption@addsubcontentsline}%
916     \ifnum\c@name caption@c@\@c@type\endcsname=\value\@c@type \else
917       \caption@Debug{Reset sub\@c@type\ counter}%
918       \expandafter\xdef\c@name caption@c@\@c@type\endcsname{%
919         \the\value\@c@type}%
920       \@stpet\@subc@type
921     \fi
922     \c@ContinuedFloat=0\relax
923     \let\caption@resetContinuedFloat\@gobble
924     \let\caption@addcontentsline\caption@kernel@addcontentsline
925     \let\caption@setfloatcapt\@firstofone
926     \caption@clearmargin
927     \caption@iflist{}{\let\caption@setlist\@gobble}%
928     \caption@setoptions{sub}%
929     \caption@setoptions{subfloat}% for subfig-package compatibility
930     \let\caption\caption@subcaption
931     \let\phantomcaption\caption@subphantom
932     \if@subfloatrow
933       \caption@Debug{Keeping \string\@makecaption}%
934     \else
935       \let\@makecaption\caption@makecaption
936     \fi
937   \fi}%
```

\if@subfloatrow

This macro tests if we are inside an subfloatrow or subfloatrow* environment.

```
938 \caption@AtBeginDocument{%
939   \caption@ifundefined\@subfloatrowtrue
940   {\newif\if@subfloatrow
941     \caption@ifundefined\subfloatrow{}%
942     {\caption@Debug{Patching subfloatrow environment}%
943       \g@addto@macro\capsubrowsettings{\@subfloatrowtrue}%
944       \g@addto@macro\killfloatstyle{%
945         \ifx\c@FRobj\c@FRsobj\@subfloatrowtrue\fi}}}%
946   {\caption@Debug{\string\if@subfloatrow is already defined}}}%
```

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

```

947 \newcommand*\caption@subcaption{%
948   \caption@checkgrouplevel{sub}\subcaption
949   \caption@star
950   {\caption@refstepcounter\@subcaptype}%
951   {\caption@dblarg{\@caption\@subcaptype}}}
```

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

```

952 \newcommand*\caption@subphantom{%
953   \caption@checkgrouplevel{sub}\phantomsubcaption
954   \caption@refstepcounter\@subcaptype}
```

`\caption@addcontentsline` **We extend `\caption@addcontentsline` so it handles sub-captions, too.**
Note: `\sf@ifpositiontop` & `\@listsubcaptions` are defined by the `subfigure` & `subfig` packages.

```

955 \let\caption@kernel@addcontentsline\caption@addcontentsline
956 \renewcommand\caption@addcontentsline[2]{%
957   \sf@ifpositiontop{\@listsubcaptions{#1}}{}%
958   \caption@kernel@addcontentsline{#1}{#2}%
959   \sf@ifpositiontop{}{\@listsubcaptions{#1}}%
960   \caption@addsubcontentslines{#1}}

961 \newcommand*\caption@addsubcontentslines[1]{%
962   \begingroup
963     \caption@subcontentslines
964   \endgroup
965   \caption@clearsubcontentslines}%

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

```

966 \newcommand*\caption@addsubcontentsline[4]{%
967   \begingroup
968   \let\label\caption@gobble \let\index\caption@gobble \let\glossary\caption@gobble
969   \protected@edef\@tempa{\endgroup
970     \noexpand\g@addto@macro\noexpand\caption@subcontentslines{%
971       \noexpand\@namedef{the#2}{\csname the#2\endcsname}%
972       \ifx\@currentHref\@undefined \else
973         \noexpand\def\noexpand\@currentHref{\@currentHref}%
974       \fi
975       \protect\caption@@@addcontentsline{#1}{#2}{#3}{#4}}}%
976   \@tempa}
```

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

```

977 \newcommand*\caption@checksubcontentslines{%
978   \ifx\caption@subcontentslines\@empty \else
979     \caption@Error{%
980       Something's wrong--perhaps a missing \protect\caption\MessageBreak
981       in the last figure or table}%
982   \caption@clearsubcontentslines
983   \fi}
```

`\caption@clearsubcontentslines` **Clear pending sub-caption list entries.**

```

984 \newcommand*\caption@clearsubcontentslines{%
985   \global\let\caption@subcontentslines\@empty}
```

```

986 \caption@AtBeginDocument*{%
987   \caption@ifundefined\sf@ifpositiontop{\let\sf@ifpositiontop@gobbletwo}{}%
988   \caption@clearsubcontentslines
989   \g@addto@macro\caption@typehook{\caption@checksubcontentslines}%
990   \AtEndDocument{\caption@checksubcontentslines}}%

```

15 Document class & Babel package support

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

```

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

```

15.2 The beamer class

```

992 \@ifclassloaded{beamer}{%
993   \caption@InfoNoLine{beamer document class}%

```

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

```

\table 994   \expandafter\let\expandafter\caption@ORI@figure
          \csname\string\figure\endcsname
995   \@namedef{\string\figure}[#1]{%
996     \caption@ORI@figure[#1]%
997     \caption@settype{figure}}
998
999   \expandafter\let\expandafter\caption@ORI@table
1000     \csname\string\table\endcsname
1001   \@namedef{\string\table}[#1]{%
1002     \caption@ORI@table[#1]%
1003     \caption@settype{table}}
1004 {}{}

```

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.

```

1005 \caption@ifundefined\scr@caption{}{%
1006   \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)

```

1007 \@nameuse{caption@frenchb}\@nameundef{caption@frenchb}

```

15.5 The frenchle/pro package

```

1008 \caption@AtBeginDocument{\caption@ifundefined\frenchTeXmods{}{%
1009   \caption@InfoNoLine{frenchle/pro package is loaded}%
1010   \let\captionfont@ORI\captionfont
1011   \let\captionlabelfont@ORI\captionlabelfont
1012   \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`.

```

1013 \caption@ifundefined\GOfrench
1014   {\let\caption@tempa\@firstofone}%
1015   {\def\caption@tempa{\g@addto@macro\GOfrench}}%
1016 \caption@tempa{%
1017   \let\captionfont\captionfont@ORI
1018   \let\captionfont@ORI\@undefined
1019   \let\captionlabelfont\captionlabelfont@ORI
1020   \let\captionlabelfont@ORI\@undefined
1021   \let\@makecaption\@makecaption@ORI
1022   \let\@makecaption@ORI\@undefined

```

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

```

1023 \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.

```

1024 \let\caption@tcORI\@tablescaption
1025 \def\@tablescaption{\caption@star\relax\caption@tcORI}%

```

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

```

1026 \let\@eatDP\@undefined
1027 \let\caption@tempa\@empty
1028 \ifx\f@ffrench\fnum@figure
1029   \l@addto@macro\caption@tempa{\let\fnum@figure\f@ffrench}%
1030 \fi
1031 \ifx\f@tfrench\fnum@table
1032   \l@addto@macro\caption@tempa{\let\fnum@table\f@tfrench}%
1033 \fi
1034 \def\f@ffrench{\ifx\listoffigures\relax\else\figurename~\thefigure\fi}%
1035 \def\f@tfrench{\ifx\listoftables\relax\else\tablename~\thetable\fi}%
1036 \caption@tempa
1037 }%
1038 }}

```

15.6 The hungarian and magyar Babel option

```

1039 \def\caption@tempa#1{%
1040   \@ifundefined{extras#1}\caption@AtBeginDocument\@firstofone{%
1041     \@ifundefined{extras#1}}{}%
1042     \caption@InfoNoLine{#1 babel option is loaded}%
1043     \expandafter\addto\csname extras#1\endcsname{%
1044       % reverse changes made by magyar.ldf
1045       \let\@makecaption\caption@makecaption
1046       \babel@save\@makecaption
1047       \caption@redefine

```

```

1048         \babel@save\@caption}%
1049     }}}
1050 \caption@tempa{hungarian}%
1051 \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.
1052 \newcommand\caption@ifpackageloaded[1]{%
1053   \@testopt{\caption@ifpackageloaded{#1}}{}}
1054 \@onlypreamble\caption@ifpackageloaded
1055 \long\def\caption@ifpackageloaded#1[#2]#3#4{%
1056   \@ifpackageloaded{#1}\@firstofone{%
1057     \caption@Debug{#1 package is not loaded (yet)\@gobble}%
1058     \caption@AtBeginDocument{%
1059       \caption@ifpackageloaded{#1}[#2]{#3}{#4}}}
1060 \@onlypreamble\caption@ifpackageloaded
1061 \newcommand\caption@ifpackageloaded[1]{%
1062   \@testopt{\caption@ifpackageloaded{#1}}{}}
1063 \@onlypreamble\caption@ifpackageloaded
1064 \long\def\caption@ifpackageloaded#1[#2]{%
1065   \@ifpackageloaded{#1}{%
1066     \caption@InfoNoLine{#1 package is loaded}%
1067     \@ifpackagelater{#1}{#2}\@firstoftwo{%
1068       \caption@Error{%
1069         For a successful cooperation we need at least version\MessageBreak
1070         `#2' of package #1,\MessageBreak
1071         but only version\MessageBreak
1072         `\'csname ver@#1.\@pkgextension\endcsname'\MessageBreak
1073         is available}%
1074       \@secondoftwo}%
1075     }{\@secondoftwo}}
1076 \@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.
1077 \newcommand*\caption@clearmargin{%
1078   \setcaptionmargin\z@
1079   \let\caption@minmargin\undefined}

1080 \caption@setbool{needfreeze}{0}
1081 \caption@AtBeginDocument*{%
1082   \caption@ifneedfreeze{%

\caption@freeze \caption@freeze
Used by the fltpage & sidecap package support.
1083   \newcommand*\caption@freeze{%

```

```

1084 \let\caption@frozen@ContinuedFloat\ContinuedFloat
1085 \def\ContinuedFloat{%
1086   \caption@withoptargs\caption@SC@ContinuedFloat}%
1087 \def\caption@SC@ContinuedFloat##1{%
1088   \caption@@freeze{\ContinuedFloat##1}%
1089   \let\caption@frozen@setcounter\setcounter
1090   \let\caption@frozen@addtocounter\addtocounter
1091   \def\setcounter####1####2{\csname c@####1\endcsname####2\relax}%
1092   \def\addtocounter####1####2{\advance\csname c@####1\endcsname ####2\relax}%
1093   \caption@frozen@ContinuedFloat##1%
1094   \let\setcounter\caption@frozen@setcounter
1095   \let\addtocounter\caption@frozen@addtocounter}%
1096 \let\caption@frozen@setup\caption@setup
1097 \def\caption@setup##1{%
1098   \caption@@freeze{\caption@setup{##1}}%
1099   \caption@frozen@setup{##1}}%
1100 \let\caption@frozen@caption\caption
1101 \def\caption{%
1102   \def\caption{%
1103     \caption@Error{%
1104       Only one \noexpand\caption can be placed in this environment}%
1105     \caption@gobble}%
1106   \@ifstar
1107   {\caption@SC@caption*}%
1108   {\let\caption@frozen@refstepcounter\caption@@refstepcounter
1109     \let\caption@@refstepcounter\caption@l@stepcounter
1110     \caption@refstepcounter\@capttype
1111     \let\caption@@refstepcounter\caption@frozen@refstepcounter
1112     \let\@currentlabel\caption@SCLabel
1113     \caption@withoptargs\caption@SC@caption}}%
1114 \long\def\caption@SC@caption##1##2{%
1115   \caption@@freeze{\caption##1{##2}}%
1116   \ignorespaces}%
1117 \let\caption@frozen@label\label
1118 \def\label{%
1119   \caption@withoptargs\caption@SC@label}%
1120 \def\caption@SC@label##1##2{%
1121   \ifx\@currentlabel\caption@SCLabel
1122     \@bsphack
1123     \caption@freeze@label{##1}{##2}%
1124     \@esphack
1125   \else
1126     \caption@frozen@label##1{##2}%
1127   \fi}%
1128 \def\caption@SCLabel{\caption@undefinedlabel}%
1129 \def\caption@freeze@label##1##2{%
1130   \caption@@freeze{\label##1{##2}}}%
1131 \global\let\caption@frozen@content\@empty
1132 \long\def\caption@@freeze{%
1133   \g@addto@macro\caption@frozen@content}%
1134 \def\caption@warmup{%
1135   \let\ContinuedFloat\caption@frozen@ContinuedFloat

```



```

1136      \let\caption@setup\caption@frozen@setup
1137      \let\caption\caption@frozen@caption
1138      \let\label\caption@frozen@label}}%

\caption@defrost \caption@defrost
1139  \newcommand*\caption@defrost{%
1140    \ifx\caption@frozen@caption\@undefined
1141      \caption@frozen@content
1142    \else
1143      \caption@Error{Internal Error:\MessageBreak
1144        \noexpand\caption@defrost in same group as \string\caption@freeze}%
1145    \fi}%

1146  }{}%
1147  \caption@undefbool{needfreeze}}

\caption@warmup \caption@warmup
1148 \let\caption@warmup\relax

```

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.)

```

1149 \caption@ifPackageLoaded{float}[2001/11/08 v1.3d]{%
1150   \@ifpackageloaded{floatrow}{%
1151     \caption@ifpackageloaded{floatrow}[2007/08/24 v0.2a]{}{}%
1152   }{}%

```

`\@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>]`.

```

1153   \let\caption@ORI@float@setevery\@float@setevery
1154   \def\@float@setevery#1{%
1155     \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.

```
1156 \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.

```
1157 \renewcommand\caption@setfloatcapt[1]{%
1158 \let\@makecaption\caption@@make
1159 \global\setbox\@floatcapt\vbox{%
1160 \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.

```
1161 \float@getstyle\float@style{#1}%
1162 \caption@setstyle*\float@style
1163 \caption@setoptions\float@style
1164 }{}%
1165 \caption@freezeHref % will be defrosted in \float@makebox
1166 \caption@ORI@float@setevery{#1}}%
```

`\caption@typehook` \LaTeX 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.

```
1167 \g@addto@macro\caption@typehook{%
1168 \expandafter\ifx\csname #1name\endcsname\relax
1169 \expandafter\let\csname #1name\expandafter\endcsname
1170 \csname fname@#1\endcsname
1171 \fi}%
```

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

```
1172 \g@addto@macro\fs@plaintop{\def\@fs@mid{\vspace\abovecaptionskip\relax}}%
1173 \g@addto@macro\fs@boxed{\def\@fs@mid{\kern\abovecaptionskip\relax}}%
```

`\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.

```
1174 \providecommand*\float@getstyle[2]{%
1175   \edef#1{%
1176     \noexpand\expandafter\noexpand\@gobblefour\noexpand\string
1177     \expandafter\expandafter\expandafter\noexpand
1178     \csname fst@#2\endcsname}%
1179   \edef#1{#1}%
1180   \caption@Debug{floatstyle{#2} = '#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@ifcaption[1]{\@secondoftwo}%
1189 % \clearcaptionsetup{boxed}% used by the floatrow package?
1190 }
```

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

```
1191 \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’.

```
1192 \caption@ifbool{ruled}{%
1193   \captionsetup[ruled]{margin=0pt,minmargin=0,slc=0}%
1194 }{%
1195   \DeclareCaptionStyle{ruled}{labelfont=bf,labelsep=space,strut=0}%
1196 }
1197 \caption@undefbool{ruled}
```

16.2 The floatflt package

```
1198 \caption@ifPackageLoaded{floatflt}[1996/02/27 v1.3]{%
```

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

```

1199 \let\caption@ORI@floatingfigure\floatingfigure
1200 \def\floatingfigure{%
1201     \caption@floatflt{figure}%
1202     \caption@ORI@floatingfigure}%

```

`\floatingtable` Same with `\floatingtable...`

```

1203 \let\caption@ORI@floatingtable\floatingtable
1204 \def\floatingtable{%
1205     \caption@floatflt{table}%
1206 %    \caption@setautoposition b%
1207     \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.

```

1208 \newcommand*\caption@floatflt[1]{%
1209     \caption@settype{#1}%
1210     \caption@clearmargin
1211     \caption@setfullparboxrestore
1212     \caption@setoptions{floating#1}}%
1213 {}

```

16.3 The `fltpage` package

```

1214 \caption@IfPackageLoaded{fltpage}[1998/10/29 v.0.3]{%
1215     \caption@setbool{needfreeze}{1}%

```

`\FP@positionLabel` Original code:

```

\newcommand{\FP@positionLabel}{%
    FP\@captype-\number\value{FP@\@captype C}-pos}

```

```

1216 \renewcommand\FP@positionLabel{%
1217     FP\FP@captype-\number\value{FP@\FP@captype C}-pos}%

```

`\FP@helpNote` Original code:

```

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

```

```

1218 \renewcommand\FP@helpNote[2]{%
1219     \begingroup % save \caption@thepage
1220     \caption@pageref{#2}%
1221     \typeout{FP#1 is inserted on page \caption@thepage!}%
1222     \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}%
}%

1223 \renewcommand*\FP@floatBegin[1]{%
1224   \def\FP@captype{#1}%
1225   \begin{lrbox}{\FP@floatCorpusBOX}%
1226   \caption@settype*{#1}%
1227   \caption@freeze
1228   \global\let\FP@LabelText\@empty
1229   \caption@ifFPrefcap
1230   {}%
1231   {\def\caption@freeze@label##1##2{%
1232     \g@addto@macro\FP@Label{\FP@label##1{##2}}}%
1233   \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}}%
}%

1234 \renewcommand*\FP@floatEnd{%
1235 \end{lrbox}%
1236 \stepcounter{FP@\FP@capttype C}%
1237 \caption@label\FP@positionLabel
1238 \FP@helpNote\FP@capttype\FP@positionLabel
1239 \edef\FP@RestoreCounter{%
1240 \noexpand\setcounter{\FP@capttype}{\the\value\FP@capttype}%
1241 \noexpand\setcounter{ContinuedFloat}{\the\value{ContinuedFloat}}}%
1242 \FP@float
1243 {\FP@positionLabel}% location label test
1244 {\begin\FP@capttype[p!}%
1245 \usebox\FP@floatCorpusBOX
1246 \let\caption@SClentry\@empty
1247 \def\caption{\caption@dblarg{\@caption\@capttype}}%
1248 \long\def\@caption##1[##2]##3{\def\caption@SClentry{##2}}%
1249 \let\FP@label\label
1250 \let\label\caption@gobble
1251 \caption@defrost
1252 \caption@ifFPlistcap
1253 {\caption@refstepcounter\@capttype
1254 \expandafter\caption@makecurrent\expandafter\@capttype
1255 \expandafter{\caption@SClentry}}%
1256 {\ifx\caption@SClentry\@empty \else
1257 \expandafter\caption@listentry\expandafter{\caption@SClentry}%
1258 \fi}%
1259 \caption@makeanchor\relax
1260 \FP@Label
1261 \end\FP@capttype}%
1262 {\FP@RestoreCounter
1263 \@ifundefined{theH\FP@capttype}{}{}%
1264 \expandafter\l@addto@macro\csname theH\FP@capttype\endcsname{.FP}}}%
1265 {\begin\FP@capttype[b!}%
1266 \let\FP@savedSetfnumCommand\caption@setfnum
1267 \def\caption@setfnum##1{%
1268 \FP@savedSetfnumCommand{##1}%
1269 \ifx\FP@guide\@empty \else
1270 \expandafter\l@addto@macro\csname fnum@##1\endcsname{\ \FP@guide}}%
1271 \fi}%
1272 \setlength\abovecaptionskip{2pt plus 2pt minus 1pt}% length above captio
1273 \setlength\belowcaptionskip{2pt plus 2pt minus 1pt}% length below captio
1274 \caption@setoptions{FP@\@capttype}%
1275 \FP@separatorCaption
1276 \caption@ifFPlistcap{}{\let\caption@addcontentsline\@gobbletwo}%
1277 \caption@defrost
1278 \end\FP@capttype}%
1279 }%

```

```

1280 }{%
1281   \let\caption@ifFPlstcap\@undefined
1282   \let\caption@ifFPrefcap\@undefined
1283 }

```

16.4 The hyperref package

```

1284 \caption@ifPackageLoaded{hyperref}[2003/11/30 v6.74m]{%
1285   % Test if hyperref has stopped early
1286   \caption@ifundefined\IfHyperBoolean{%
1287     \caption@set@bool\caption@ifhyp@stoppeearly0%
1288     \caption@ifundefined\H@refstepcounter
1289     {\caption@set@bool\caption@ifhyp@stoppeearly1}{%
1290       \caption@ifundefined\hyper@makecurrent
1291       {\caption@set@bool\caption@ifhyp@stoppeearly1}{%
1292         \caption@ifundefined\measuring@true
1293         {\caption@set@bool\caption@ifhyp@stoppeearly1}{}}}%
1294   }{%
1295     \def\caption@ifhyp@stoppeearly{\IfHyperBoolean{stoppeearly}}%
1296   }%
1297   \caption@ifhyp@stoppeearly{% hyperref has stopped early
1298     \caption@InfoNoLine{%
1299       Hyperref support is turned off\MessageBreak
1300       because hyperref has stopped early}%
1301   }{%
1302     \g@addto@macro\caption@prepareslc{\measuring@true}%

```

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

```

1303   \renewcommand*\caption@@refstepcounter{\H@refstepcounter}%

```

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

```

1304   \renewcommand*\caption@makecurrent[2]{%
1305     \caption@makecurrentHref{#1}%
1306     \caption@Debug{hyperref current=\@currentHref}%
1307     \caption@getttitle{#2}}%
1308   \newcommand*\caption@makecurrentHref{\hyper@makecurrent}%

```

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

```

1309   \renewcommand\caption@makeanchor[1]{%
1310     \caption@Debug{hyperref anchor: \@currentHref}%
1311     % If we cannot have nesting, the anchor is empty.
1312     \ifHy@nesting
1313       \expandafter\hyper@@anchor\expandafter{\@currentHref}{#1}%
1314     \else
1315       \Hy@raisedlink{%
1316         \expandafter\hyper@@anchor\expandafter{\@currentHref}{\relax}%
1317       }#1%
1318     \fi}%
1319   \g@addto@macro\caption@prepareslc{\let\caption@makeanchor\@firstofone}%

```

The hypcap option

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

```
1320 \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`.

```
1321 \def\caption@start{\caption@ifhypcap\caption@start@relax}%
1322 \def\caption@start@{%
```

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

```
1323 \caption@makestart\@captype
1324 \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.)

```
1325 \global\@capstarttrue
1326 \let\hc@currentHref\@currentHref
1327 \def\caption@makecurrentHref##1{%
1328 \global\@capstartfalse
1329 \global\let\@currentHref\hc@currentHref}%
```

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

```
1330 \let\caption@makeanchor\@firstofone
1331 }%
```

`\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.

```
1332 \newcommand*\caption@makestart[1]{%
1333 \begingroup
1334 \Hy@hypertexnamesfalse
1335 % \gdef\@currentHlabel{}%
1336 \hyper@makecurrent{#1.caption}%
1337 \endgroup
1338 \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.

```

1339 \newcommand*\caption@startanchor[1]{%
1340   \ifvmode\begin{group}
1341     \caption@Debug{hypcap anchor: #1 (vertical mode)}%
1342     \@tempdima\prevdepth
1343     \nointerlineskip
1344     \vspace*{-\caption@hypcapspace}%
1345     \caption@anchor{#1}%
1346     \vspace*{\caption@hypcapspace}%
1347     \prevdepth\@tempdima
1348   \endgroup\else
1349     \caption@Debug{hypcap anchor: #1 (horizontal mode)}%
1350     \caption@anchor{#1}%
1351   \fi}%

```

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

```

1352 \newcommand*\caption@anchor[1]{%
1353   \ifmeasuring@ \else
1354     \caption@raisedlink{\hyper@anchorstart{#1}\hyper@anchorend}%
1355   \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`.)

```

1356 \ifx\HyperRaiseLinkLength\@tempdima
1357   \def\caption@raisedlink#1{\ifvmode#1\else\Hy@raisedlink{#1}\fi}%
1358 \else
1359   \let\caption@raisedlink\Hy@raisedlink
1360 \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.

```

1361 \def\caption@@start{%
1362   \caption@ifundefined\hc@currentHref{%
1363     \caption@Warning{%
1364       The option ‘hypcap=true’ will be ignored for this\MessageBreak
1365       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`.

```

1366 \def\caption@freezeHref{%
1367   \let\caption@ORI@start\caption@start
1368   \def\caption@start{\let\caption@start\caption@ORI@start}%

```

```

1369 % \let\caption@ORI@@start\caption@@start
1370 % \l@addto@macro\caption@subtyhook{%
1371 % \let\caption@@start\caption@ORI@@start}%

1372 \global\let\caption@currentHref\@undefined
1373 \def\caption@@start{\global\let\caption@currentHref\@currentHref}%

1374 \let\caption@ORI@setfloatcapt\caption@setfloatcapt
1375 \renewcommand*\caption@setfloatcapt{%
1376 \ifx\caption@currentHref\@undefined \else
1377 \let\caption@makeanchor\@firstofone
1378 \fi
1379 \caption@ORI@setfloatcapt}}%

```

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

```

1380 \def\caption@defrostHref{%
1381 \ifx\caption@currentHref\@undefined \else
1382 \caption@startanchor\caption@currentHref
1383 \global\let\caption@currentHref\@undefined
1384 \fi}%

```

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

```

1385 \caption@ifundefined\HyOrg@float@makebox{}{%
1386 \caption@Debug{%
1387 Redefining \noexpand\float@makebox (again)\@gobble}%
1388 \let\caption@ORI@float@makebox\float@makebox % save for compatibility mode
1389 \renewcommand\float@makebox[1]{%
1390 \HyOrg@float@makebox{#1\relax \caption@defrostHref}}%
1391 }%

1392 }}{}

```

16.5 The hypcap package

```

1393 \caption@ifPackageLoaded{hypcap}{% v1.0
1394 \ifx\caption@start\relax \else % hyperref hasn't stopped early

```

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

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

```

1395 \let\caption@ORI@capstart\capstart % save for compatibility mode
1396 \caption@ifundefined\capstarttrue % check for v1.10 of hypcap package
1397 {\def\capstart{\caption@start@}}%
1398 {\def\capstart{\ifcapstart\caption@start@\fi}}%
1399 \let\caption@start\relax
1400 \let\caption@@start\relax

```

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

```

1401 \caption@set@bool\caption@ifhypcap 1%
1402 \renewcommand*\caption@hypcapspace{\hypcapspace}%

```

```
1403 \fi}{}
```

16.6 The listings package

```
1404 \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.

```
1405 \let\caption@ORI@lst@MakeCaption\lst@MakeCaption
1406 \def\lst@MakeCaption#1{% #1 is ‘t’ or ‘b’
1407 \begingroup
```

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

```
1408 \ifdim\hsize>\linewidth
1409 \hsize\linewidth
1410 \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`.)

```
1411 \caption@setposition{#1}%
1412 \caption@iftop{%
1413 \@tempdima\belowcaptionskip
1414 \belowcaptionskip\abovecaptionskip
1415 \abovecaptionskip\@tempdima}{}%
```

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

```
1416 \caption@setup{rule=0}%
```

Afterwards we set the local ‘lstlisting’ options.

```
1417 \caption@setoptions{lstlisting}%
```

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

```
1418 \caption@setautoposition{#1}%
```

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

```
1419 \caption@begin{lstlisting}%
1420 \caption@ORI@lst@MakeCaption{#1}%
1421 \caption@end
1422 \endgroup}%
```

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

```
\lst@maketitle 1423 \def\lst@makecaption{\caption@starfalse\@makecaption}%
1424 \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.

```
1425 \providecommand*\ext@lstlisting{lol}%
1426 }{}
```

16.7 The longtable package

```

\LTcapttype \LTcapttype is preset to table.
1427 \providecommand*\LTcapttype{table}

1428 \caption@ifPackageLoaded{longtable}[1995/05/24 v3.14]{%
1429   \RequirePackage{ltcaption}[2007/09/01]%
1430   \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.

1431 \caption@AtBeginDocument{%
1432   \let\caption@ORI@LT@array\LT@array
1433   \renewcommand*\LT@array{%

\captionsetup for longtable:
1434   \global\let\caption@opt@@longtable\@undefined
1435   \def\captionsetup{%
1436     \noalign\bgroup
1437     \ifstar\captionsetup\captionsetup}% gobble *
1438   \def\captionsetup##1{\LT@captionsetup{##1}\egroup}%
1439   \def\LT@captionsetup##1{%
1440     \captionsetup@starttrue\caption@setup@options[@longtable]{##1}%
1441     \global\let\caption@opt@@longtable\caption@opt@@longtable}%

\captionabove & \captionbelow for longtable: (KOMA-Script document class)
1442   \def\captionabovetrue{\LT@captionsetup{position=t}}%
1443   \def\captionabovefalse{\LT@captionsetup{position=b}}%

\captionlistentry for longtable:
1444   \def\captionlistentry{%
1445     \noalign\bgroup
1446     \ifstar{\egroup\LT@captionlistentry}% gobble *
1447     {\egroup\LT@captionlistentry}}%
1448   \def\LT@captionlistentry##1{%
1449     \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

1450 %   \caption@ifhyperc{%
1451 %     \let\caption@ORI@hyper@makecurrent\hyper@makecurrent
1452 %     \def\hyper@makecurrent##1{%
1453 %       \let\hyper@makecurrent\caption@ORI@hyper@makecurrent
1454 %       \caption@makestart{##1}%
1455 %       \let\Hy@LT@currentHlabel\@currentHlabel
1456 %       \let\Hy@LT@currentHref\@currentHref
1457 %       \def\hyper@makecurrent###1{%
1458 %         \let\@currentHlabel\Hy@LT@currentHlabel
1459 %         \let\@currentHref\Hy@LT@currentHref}}%
1460 %       \let\caption@ORI@ContinuedFloat\ContinuedFloat

```

```

1461 %      \def\ContinuedFloat{\noalign{%
1462 %          \gdef\caption@setContinuedFloat{%
1463 %              \let\caption@resetContinuedFloat\@gobble}%
1464 %      \def\caption@setoptions###1{%
1465 %          \g@addto@macro\caption@setContinuedFloat{%
1466 %              \caption@setoptions{###1}}}%
1467 %      \let\@capttype\LTcapttype
1468 %      \caption@ORI@ContinuedFloat}}%
1469 %  }{%
1470 %      \def\ContinuedFloat{\noalign{%
1471 %          \caption@Error{%
1472 %              \noexpand\ContinuedFloat inside longtables\MessageBreak
1473 %              is only available with 'hyccap=true'}}}%
1474 %  }%
1475 %      \global\let\caption@setContinuedFloat\@empty
1476 %      \def\ContinuedFloat{\noalign{%
1477 %          \caption@Error{\noexpand\ContinuedFloat outside float}}}%
1478 %      \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}`:

```

1479 \long\def\LT@c@ption#1[#2]#3{%
1480   \LT@makecaption#1{\csname fnun@\LTcapttype\endcsname}{#3}%
1481   \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:

```

1482 \renewcommand\LT@makecaption[3]{%
1483   \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`.)

```
1484      \caption@settype*\LTcaptype
1485      \ifdim\LTcapwidth=4in \else
1486        \setcaptionwidth\LTcapwidth
1487      \fi
1488      \caption@setoptions{longtable}%
1489 %      \caption@setContinuedFloat
1490      \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!

```
1491      \caption@setautoposition{\ifcase\LT@rows t\else b\fi}%
```

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

```
1492      \caption@startrue#1\caption@starfalse
1493      \caption@resetContinuedFloat\LTcaptype
1494      \caption@begin\LTcaptype
1495      \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...)

```
1496      \vskip-\ht\strutbox
```

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

```
1497      \caption@iftop{\vskip\belowcaptionskip}{\vskip\abovecaptionskip}%
1498      \caption@@make{#2}{#3}\endgraf
1499      \caption@iftop{\vskip\abovecaptionskip}{\vskip\belowcaptionskip}%
1500      \caption@end}%%
```

```
1501 }{ }
```

16.8 The picinpar package

```
1502 \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.

```
1503   \long\def\figwindow[#1,#2,#3,#4] {%
1504     \caption@window{figure}%
1505     \caption@setoptions{figwindow}%
1506     \begin{window}[#1,#2,{#3},\caption@wincaption{#4}] }%
```

```

1507 \long\def\tabwindow[#1,#2,#3,#4] {%
1508   \caption@window{table}%
1509   \caption@setoptions{tabwindow}%
1510   \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.**

```

1511 \newcommand*\caption@window[1]{%
1512   \let\@makecaption\caption@@make
1513   \caption@setautoposition b%
1514   \caption@settype{#1}%
1515   \caption@clearmargin
1516   \caption@setfullparboxrestore}%

```

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

```

1517 \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`.

```

1518   \ifdim\picwd=\z@
1519     \let\caption@makecurrent\@gobbletwo
1520     \let\caption@@start\relax
1521     \caption@prepareslc
1522   \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.

```

1523   \edef\@tempa{\expandafter\noexpand\@car#1\@nil}%
1524   \if\@tempa*%
1525     \let\@tempa\@firstofone
1526   \else\if\@tempa[%
1527     \let\@tempa\@firstofone
1528   \else
1529     \let\@tempa\@empty
1530   \fi\fi
1531   \expandafter\caption\@tempa{#1}}%

```

```

1532 }{}

```

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 `\@captive`, 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.

```

1533 \newcommand*\piccaptiontype[1]{\def\@piccaptiontype{#1}}

```

```

1534 \caption@ifpackageloaded{picins}{%

```

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

```
1535 \caption@ifundefined\@piccaptiontype{%
1536   \caption@iftype{%
1537     \let\@piccaptiontype\@capttype
1538   }{%
1539     \def\@piccaptiontype{figure}%
1540   }%
1541 }{}%
1542 \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.

```
1543 \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.

```
1544 \let\caption@ORI@make@piccaption\make@piccaption
1545 \def\make@piccaption{%
1546   \let\caption@ORI\caption
1547   \long\def\caption[##1]##2{%
1548     \caption@freezeHref % will be defrosted in \ivparpic
1549     \caption@settype\@piccaptiontype
1550 % \ifnum\c@piccaptionpos>2\relax
1551     \caption@clearmargin
1552 % \else
1553 % \captionwidth\z@ % do not use "width=" setting
1554 % \fi
1555     \caption@setfullparboxrestore
1556     \caption@setoptions{parpic}%
1557     \caption@setautoposition b%
```



```

1558     \expandafter\expandafter\expandafter\caption@ORI
1559     \expandafter\expandafter\expandafter[%
1560     \expandafter\expandafter\expandafter{%
1561     \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

1562     \caption@ORI@make@piccaption
1563     \let\caption\caption@ORI}%

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

1564     \let\caption@ORI@ivparpic\ivparpic
1565     \def\ivparpic(#1,#2)(#3,#4)[#5][#6]#7{%
1566     \let\caption@ORI@noindent\noindent
1567     \def\noindent{%
1568     \caption@defrostHref
1569     \let\noindent\caption@ORI@noindent
1570     \noindent}%
1571     \caption@ORI@ivparpic(#1,#2)(#3,#4)[#5][#6]{#7}%
1572     \let\noindent\caption@ORI@noindent}%

1573 }{%
1574     \let\piccaptiontype\@undefined
1575 }

```

16.10 The rotating package

```

1576 \caption@ifPackageLoaded{rotating}[1995/08/22 v2.10]{%

\rotcaption Make \rotcaption* work.

1577 \def\rotcaption{\let\makecaption\@makerotcaption\caption}%
1578 % \let\@rotcaption\@undefined

\rotcaptionof Make \rotcaptionof(*) work.

1579 \def\rotcaptionof{%
1580     \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.)

```

1581 \long\def\@makerotcaption#1#2{%
1582     \ifdim\captionwidth=\z@
1583         \setcaptionwidth{.8\textheight}%
1584         \caption@slc{#1}{#2}{.8\vsizer}%
1585         \let\caption@makerot\caption@@make
1586         \caption@clearmargin
1587 %         \long\def\caption@parbox##1##2{\hbox{\hsize=.8\textheight\relax##2}}%
1588 %         (not needed because \rotatebox uses an \hbox anyway)
1589         \let\caption@parbox\@secondoftwo}%
1590         \caption@set@bool\caption@ifslc0% been there, done that
1591     \fi

1592     \rotatebox{90}{\caption@makerot{#1}{#2}}%
1593     \nobreak\hspace{12pt}}%

1594 \newcommand\caption@makerot[2]{%
1595     \begin{minipage}\captionwidth\caption@@make{#1}{#2}\end{minipage}}%

1596 {}

```

16.11 The sidecap package

```

1597 \caption@ifPackageLoaded{sidecap}[2003/06/06 v1.6f]{%
1598     \caption@setbool{needfreeze}{1}%

```

`\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`.

```

1599 \let\caption@ORI@SC@zfloat\SC@zfloat
1600 \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.

```

1601     \caption@ORI@SC@zfloat{#1}{#2}{#3}[#4]%
1602     \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.

```

1603     \caption@settype*{#2}%
1604     \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.

```

1605     \let\SC@label\label}%

1606 \providecommand*\SC@RestoreCommands{%
1607     \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.

```
1608 \let\caption@ORI@endSC@FLOAT\endSC@FLOAT
1609 \def\endSC@FLOAT{%
```

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

```
1610 \let\caption@ORI@settype\caption@settype
1611 \def\caption@settype##1{% will be done in \@xfloat
1612 \caption@ORI@settype*{##1}% do not change \@currentlabel
1613 \caption@setSC@justify
1614 %%% \caption@setoptions{SCfloat}%
1615 \caption@setoptions{SC\@capttype}%
1616 \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

```
1617 \def\caption@setSC@justify{%
1618 \caption@clearmargin
1619 \ifx\SC@justify\@empty \else
1620 \let\caption@hj\SC@justify
1621 \let\SC@justify\@empty
1622 \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!

```
1623 \let\caption\SC@orig@caption
1624 \def\SC@orig@caption[##1]##2{\caption@defrost}%
```

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

```
1625 \caption@setSC@justify % for compatibility mode
1626 \caption@ORI@endSC@FLOAT}%
```

```
1627 }{ }
```

16.12 The subfigure package

```
1628 \caption@ifPackageLoaded{subfigure}[2002/01/23 v2.1]{%
```

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

```
1629 \def\sF@ifpositiontop{%
1630 \ifx\@capttype\@undefined
1631 \expandafter\@gobbletwo
1632 \else\ifx\@capttype\relax
1633 \expandafter\expandafter\expandafter\@gobbletwo
```

```

1634 \else
1635 \expandafter\expandafter\expandafter\sfi@if@position@top
1636 \fi\fi}

1637 \def\sfi@if@position@top{%
1638 \ifundefined{if\@capttype topcap}%
1639 {\@gobbletwo}%
1640 {\@nameuse{if\@capttype topcap}%
1641 \expandafter\@firstoftwo
1642 \else
1643 \expandafter\@secondoftwo
1644 \fi}}

1645 {}{}

```

16.13 The supertabular and xtab packages

```

1646 \caption@if@PackageLoaded{supertabular}[2002/07/19 v4.1e]{%
\tablecaption Make \topcaption* and \bottomcaption* work.
1647 \renewcommand*\tablecaption{%
1648 \caption@star
1649 {\refstepcounter{table}}}%
1650 {\caption@dblarg{\@xtablecaption}}}%

\@xtablecaption Make \nameref and \autoref work.
1651 \let\caption@ORI@xtablecaption\@xtablecaption
1652 \long\def\@xtablecaption[#1]#2{%
1653 \caption@getttitle{#2}%
1654 \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}

1655 \long\def\ST@caption#1[#2]#3{\par%
1656 \caption@settype*{#1}%
1657 \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.

1658 \def\caption@fixposition{%
1659 \caption@setposition{\if@topcaption t\else b\fi}}%

```

```

1660     \caption@beginex{#1}{#2}{#3}%
1661     \caption@boxrestore
1662     \caption@normalsize
1663     \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
1664     \caption@end}%

1665 {}{}

1666 \caption@ifPackageLoaded{xtab}[2000/04/09 v2.3]{%

\tablecaption Make \topcaption* and \bottomcaption* work.
1667     \renewcommand*\tablecaption{%
1668         \caption@star
1669         {\refstepcounter{table}}}%
1670         {\caption@dblarg{\@xtablecaption}}}%

\@xtablecaption Make \nameref and \autoref work.
1671     \let\caption@ORI@xtablecaption\@xtablecaption
1672     \long\def\@xtablecaption[#1]#2{%
1673         \caption@getttitle{#2}%
1674         \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}}

1675 \long\def\ST@caption#1[#2]#3{\par%
1676     \caption@settype*{#1}%
1677     \caption@setoptions{xtabular}%
1678     \def\caption@fixposition{%
1679         \caption@setposition{\if@topcaption t\else b\fi}}%
1680     \@initisotab
1681     \caption@beginex{#1}{#2}{#3}%
1682     \caption@boxrestore
1683     \caption@normalsize
1684     \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
1685     \caption@end
1686     \global\advance\ST@pageleft -\PWSTcapht
1687     \ST@trace\tw@{Added caption. Space left for xtabular: \the\ST@pageleft}}%

1688 {}{}

```

16.14 The threeparttable package

1689 \caption@ifpackageloaded{threeparttable}[2003/06/13 v3.0]{%

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

```
1690 \let\caption@ORI@threeparttable\threeparttable
1691 \renewcommand*\threeparttable{%
1692   \caption@settype{table}%
1693   \caption@setposition a% ?
1694   \caption@clearmargin
1695   \caption@setoptions{threeparttable}%
1696   \caption@ORI@threeparttable}%
```

\measuredfigure Same here...

```
1697 \let\caption@ORI@measuredfigure\measuredfigure
1698 \renewcommand*\measuredfigure{%
1699   \caption@settype{figure}%
1700   \caption@setposition a% ?
1701   \caption@clearmargin
1702   \caption@setoptions{measuredfigure}%
1703   \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}
```

```
1704 \def\TPT@caption#1[#2]#3{%
1705   \gdef\TPT@docapt{%
1706     \global\let\TPT@docapt\@undefined
1707     \caption@setautoposition\caption@TPT@position
1708     \TPT@LA@caption{#1}[{#2}]{#3}}%
1709   \ifx\TPT@hsize\@empty
1710     \let\label\TPT@gatherlabel % Bug: does not work for measuredfigures
1711     \gdef\caption@TPT@position{t}%
1712     \g@addto@macro\TPT@docapt\caption@TPT@eatvskip
1713   \else
1714     \def\caption@TPT@position{b}%
1715     \TPT@docapt
1716   \fi
1717   \ignorespaces}%
1718   %\newcommand*\caption@TPT@eatvskip{\vskip-.2\baselineskip}%
1719   \def\caption@TPT@eatvskip#1\vskip{#1\@tempdima=}%
1720 }
```

16.15 The wrapfig package

1721 \caption@ifpackageloaded{wrapfig}[2003/01/31 v3.6]{%

`\wrapfloat` First of all we make the `wrapfig` package independent from the package load order regarding the float package. (`\wrapfloat` will usually not be used when used with re-styled floats.)

```
1722 \renewcommand*\wrapfloat[1]{%
1723   \def\@captive{#1}%
1724   \@ifundefined{fst@#1}{}{%
1725     \@nameuse{fst@#1}%
1726     \def\WF@floatstyhook{\let\@currbox\WF@box
1727       \global\setbox\WF@box\float@makebox{\wd\WF@box}}}%
1728   \@ifnextchar[\WF@wr{\WF@wr[]}}
```

`\WF@rapt` Original code:

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

Our code has `\WF@captionstyhook` in addition:

```
1729 \def\WF@rapt[#1]#2{% final two args: #1 = overhang, #2 = width,
1730   \gdef\WF@ovh{#1}% hold overhang for later, when \width is known
1731   \global\setbox\WF@box\top\bggroup \setlength\hsize{#2}%
1732   \expandafter\WF@captionstyhook\expandafter{\@captive}% <= new
1733   \ifdim\hsize>\z@ \@parboxrestore \else
1734   \setbox\z@\hbox\bggroup \let\wf@@caption\caption \let\caption\wf@caption
1735   \ignorespaces \fi}%

```

`\WF@captionstyhook` We place our `hyperref` anchor here, apply the ‘wrap’ options etc.

```
1736 \def\WF@captionstyhook#1{%
1737   \let\@captive\@undefined
1738   \caption@ifundefined\@float@setevery{}{\@float@setevery{#1}}%
1739   \caption@settype{#1}%
1740   \caption@clearmargin
1741   %%% \caption@setoptions{wrapfloat}%
1742   \caption@setoptions{wrap#1}}
1743 {} }
```

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