

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

Axel Sommerfeldt

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

2011/08/06

## Abstract

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

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

## User manuals

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

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

---

<sup>\*</sup>This package has version number v3.2c, last revised 2011/09/30.

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

# Contents

<b>1 Identification</b>	<b>4</b>
<b>2 Loading the kernel</b>	<b>4</b>
<b>3 Check against incompatible document classes</b>	<b>4</b>
<b>4 Check against incompatible packages</b>	<b>4</b>
<b>5 Declaration of options</b>	<b>4</b>
5.1 Options for figure and table . . . . .	4
5.2 Miscellaneous options . . . . .	5
5.3 caption v1.x compatibility options . . . . .	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
<b>6 <math>\mathcal{A}\mathcal{M}\mathcal{S}</math> &amp; SMF document classes support</b>	<b>7</b>
<b>7 KOMA-Script document classes support</b>	<b>8</b>
<b>8 Processing of options</b>	<b>10</b>
<b>9 <code>\caption</code>, <code>\@caption</code>, and <code>\@makecaption</code></b>	<b>10</b>
<b>10 <code>\captionof</code> and <code>\captionlistentry</code></b>	<b>19</b>
<b>11 <code>\captionbox</code></b>	<b>22</b>
<b>12 <code>\ContinuedFloat</code></b>	<b>22</b>
<b>13 Internal helpers</b>	<b>24</b>
<b>14 Support for sub-captions</b>	<b>26</b>
<b>15 Document class &amp; Babel package support</b>	<b>29</b>
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

<b>16 Package support</b>	<b>31</b>
16.1 The float package . . . . .	33
16.2 The floatflt package . . . . .	36
16.3 The fltpage package . . . . .	36
16.4 The hyperref package . . . . .	39
16.5 The hypcap package . . . . .	43
16.6 The listings package . . . . .	43
16.7 The longtable package . . . . .	44
16.8 The picinpar package . . . . .	47
16.9 The picins package . . . . .	48
16.10 The rotating package . . . . .	50
16.11 The sidecap package . . . . .	51
16.12 The subfigure package . . . . .	52
16.13 The supertabular and xtab packages . . . . .	53
16.14 The threeparttable package . . . . .	54
16.15 The wrapfig package . . . . .	55

## 1 Identification

```
1 \NeedsTeXFormat{LaTeX2e}[1994/12/01]
2 \ProvidesPackage{caption}[2011/09/30 v3.2c 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[2]{%
41   \caption@setname{#1}{#2}%
42   \begingroup
43     \caption@ifundefined\language\language{}{%
44       \ifundefined{captions\language}{}{%
45         \expandafter\g@addto@macro\csname captions\language\endcsname
46           {\caption@setname{#1}{#2}}}%
47       \endgroup}

48 \newcommand*\caption@setname[2]{\@namedef{#1name}{#2}}
49 \caption@AtBeginDocument{\let\caption@SetName\caption@setname}

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

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

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

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

```

## 5.2 Miscellaneous options

```

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

\caption@setparboxrestore \caption@setparboxrestore{<partial or full>}
66 \newcommand*\caption@setparboxrestore[1]{%
67   \caption@ifinlist{#1}{full}{%
68     \caption@setfullparboxrestore
69   }{\caption@ifinlist{#1}{default,light,partial}{%
70     \let\caption@parboxrestore\@secondoftwo
71   }}%
72   \caption@Error{Undefined parboxrestore `#1'}%
73 }

```

```

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

76 \DeclareCaptionOption{parboxrestore}{\caption@setparboxrestore{#1}}
77 \captionsetup{parboxrestore=default}

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

```

### 5.3 caption v1.x compatibility options

```

83 \DeclareCaptionOption{compatibility}[1]{\caption@setbool{compatibility}{#1}}
84 \@onlypreamble@key{caption}{compatibility}

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

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

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

```

```

119 \caption@setbool{ruled}{0}
120 \DeclareCaptionOptionNoValue*{ruled}{\caption@setbool{ruled}{1}}

```

## 5.4 caption2 v2.x compatibility options

```

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

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

128 \DeclareCaptionOptionNoValue*{ignoreLTcapwidth}{%
129   \caption@WarningNoLine{Obsolete option 'ignoreLTcapwidth' ignored}}

```

## 5.5 Obsolete caption v3.0 options

```

130 \DeclareCaptionOption*{caption}{%
131   \caption@setbool{temp}{#1}%
132   \caption@ifbool{temp}{}{%
133     \caption@Error{%
134       The package option 'caption=#1' is obsolete.\MessageBreak
135       Please pass this option to the subfig package instead\MessageBreak
136       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.

```

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

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

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

```

## 5.7 hyperref package support options

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

```

146 \DeclareCaptionOption{hypcap}[1]{\caption@setbool{hypcap}{#1}}
147 \DeclareCaptionOption{hypcapSPACE}{\def\caption@hypcapSPACE{#1}}

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

```

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

```

149 \caption@ifamsclass{%
150   \caption@InfoNoLine{AMS or SMF document class}%

```

```

151 \setlength\belowcaptionskip{0pt}% set to 12pt by AMS class
152 }

```

## 7 KOMA-Script document classes support

```

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

157 \if@tablecaptionabove
158 \@tablecaptionabovetrue
159 \else
160 \@tablecaptionabovefalse
161 \fi

\onelinecaptionstrue
\onelinecaptionfalse
162 \g@addto@macro\onelinecaptionstrue{\let\caption@ifslc\@firstoftwo}
163 \g@addto@macro\onelinecaptionfalse{\let\caption@ifslc\@secondoftwo}

164 \ifonelinecaptions
165 \onelinecaptionstrue
166 \else
167 \onelinecaptionfalse
168 \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.
169 \g@addto@macro\@captionabovetrue{\let\caption@position\@firstoftwo}
170 \g@addto@macro\@captionabovefalse{\let\caption@position\@secondoftwo}

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

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

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

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

185 \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.)

```

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

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

```

`\setcapmargin`

```

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

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

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

226 \ifx\cap@margin\relax \else
227   \caption@setcapmargin
228 \fi

229 }

```

## 8 Processing of options

```
230 \caption@SetupOptions{caption}{\caption@setkeys{#1}{#2}}
231 \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.

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

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

```
241 \newcommand*\caption@star[2]{%
242   \@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<sup>A</sup>T<sub>E</sub>X PR latex/2472](#).)

```
243 \long\def\caption@@caption#1[#2]#3{%
244   \ifcaption@star \else
245     \caption@prepareanchor{#1}{#2}%
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}
```

`\caption@prepareanchor`

```

262 \newcommand*\caption@prepareanchor[2]{%
263   \caption@makecurrent{#1}{#2}%
264   \caption@ifhypcap\caption@@start{}}

```

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

We do basically the same as the original code (from the standard L<sup>A</sup>T<sub>E</sub>X document classes), but take care of the `position=` setting and use `\caption@@make` from the `caption` kernel to finally typeset the caption.

```

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

```

273 \newcommand*\caption@redefine{}
274 \g@addto@macro\caption@redefine{%
275   \caption@setbool{incompatible}{0}%
276   \caption@CheckCommand\caption{%
277     % ltfloat.dtx [2002/10/01 v1.1v LaTeX Kernel (Floats)]
278     \def\caption{%
279       \ifx\@capttype\@undefined
280         \@latex@error{\noexpand\caption outside float}\@ehd
281         \expandafter\@gobble
282       \else
283         \refstepcounter\@capttype
284         \expandafter\@firstofone
285       \fi
286       {\@dblarg{\@caption\@capttype}}%
287     }%
288   \caption@CheckCommand\caption{%
289     % beamerbaselocalstructure.sty,v 1.53 2007/01/28 20:48:21 tantau
290     \def\caption{
291       \ifx\@capttype\@undefined
292         \@latex@error{\noexpand\caption outside figure or table}\@ehd
293         \expandafter\@gobble
294       \else
295         \refstepcounter\@capttype
296         \expandafter\@firstofone
297       \fi
298       {\@dblarg{\@caption\@capttype}}%
299     }%
300   \caption@CheckCommand\caption{%
301     % float.sty [2001/11/08 v1.3d Float enhancements (AL)]
302     \renewcommand\caption{%

```

```

303     \ifx\@capttype\@undefined
304         \@latex@error{\noexpand\caption outside float}\@ehd
305         \expandafter\@gobble
306     \else
307         \refstepcounter\@capttype
308         \let\@tempf\@caption
309         \expandafter\ifx\csname @float@c@\@capttype\endcsname\relax\else
310             \expandafter\expandafter\let
311                 \expandafter\@tempf\csname @float@c@\@capttype\endcsname
312         \fi
313     \fi
314     \@dblarg{\@tempf\@capttype}}}%
315 \caption@CheckCommand\caption{%
316 % hyperref.sty [2007/02/27 v6.75t Hypertext links for LaTeX]
317 % hyperref.sty [2007/04/09 v6.76a Hypertext links for LaTeX]
318 % hyperref.sty [2007/06/12 v6.76h Hypertext links for LaTeX]
319 \def\caption{%
320     \ifx\@capttype\@undefined
321         \@latex@error{\noexpand\caption outside float}\@ehd
322         \expandafter\@gobble
323     \else
324         \H@refstepcounter\@capttype
325         \@ifundefined{fst@\@capttype}{%
326             \let\Hy@tempa\@caption
327         }{%
328             \let\Hy@tempa\Hy@float@caption
329         }%
330         \expandafter\@firstofone
331     \fi
332     {\@dblarg{\Hy@tempa\@capttype}}%
333 }%
334 \caption@CheckCommand\caption{%
335 % hyperref.sty [2007/08/05 v6.76j Hypertext links for LaTeX]
336 \def\caption{%
337     \ifx\@capttype\@undefined
338         \@latex@error{\noexpand\caption outside float}\@ehd
339         \expandafter\@gobble
340     \else
341         \H@refstepcounter\@capttype
342         \let\Hy@tempa\@caption
343         \@ifundefined{float@caption}{%
344             \expandafter\ifx\csname @float@c@\@capttype\endcsname\float@caption
345                 \let\Hy@tempa\Hy@float@caption
346             \fi
347         }%
348         \expandafter\@firstofone
349     \fi
350     {\@dblarg{\Hy@tempa\@capttype}}%
351 }%
352 }%
353 \caption@CheckCommand\caption{%
354 % memhfixc.sty [2010/08/17 v1.15 nameref/hyperref package fixes for memoir cl
355 % \let\m@moldhycaption\caption

```

```

356 \renewcommand{\caption}{\donemaincaptiontrue\m@moldhycaption}}%
357 \caption@ifcheckcommand{}{%
358 \caption@ifinfooneline{%
359 Incompatible package detected (regarding \string\caption).\MessageBreak
360 \string\caption\space=\space\meaning\caption}%
361 \caption@setbool{incompatible}{1}}%
362 \caption@checkcommand\caption{%
363 % ltfloat.dtx [2002/10/01 v1.1v LaTeX Kernel (Floats)]
364 \long\def\caption#1[#2]#3{%
365 \par
366 \addcontentsline{\csname ext@#1\endcsname}{#1}%
367 {\protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}}%
368 \begingroup
369 \@parboxrestore
370 \if@minipage
371 \@setminipage
372 \fi
373 \normalsize
374 \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
375 \endgroup}}%
376 \caption@checkcommand\caption{%
377 % beamerbaselocalstructure.sty,v 1.53 2007/01/28 20:48:21 tantau
378 \long\def\caption#1[#2]#3{% second argument ignored
379 \par\nobreak
380 \begingroup
381 \@parboxrestore
382 \if@minipage
383 \@setminipage
384 \fi
385 \beamer@makecaption{#1}{\ignorespaces #3}\par\nobreak
386 \endgroup}}%
387 \caption@checkcommand\caption{%
388 % magyar.ldf [2005/03/30 v1.4j Magyar support from the babel system]
389 \long\def\caption#1[#2]#3{%
390 \csname par\endcsname
391 \addcontentsline{\csname ext@#1\endcsname}{#1}%
392 {\protect\numberline{\csname the#1\endcsname.}{\ignorespaces #2}}%
393 \begingroup
394 \@parboxrestore
395 \if@minipage
396 \@setminipage
397 \fi
398 \normalsize
399 \@makecaption{\csname fnum@#1\endcsname}%
400 {\ignorespaces #3}\csname par\endcsname
401 \endgroup}}%
402 % \caption@checkcommand\float@caption{%
403 % float.sty [2001/11/08 v1.3d Float enhancements (AL)]
404 % \long\def\float@caption#1[#2]#3{%
405 % \addcontentsline{\@nameuse{ext@#1}}{#1}%
406 % {\protect\numberline{\@nameuse{the#1}}{\ignorespaces #2}}
407 % \global\setbox\@floatcapt\vbox\bgroup\@parboxrestore

```

```

408 %         \normalsize\@fs@capt{\@nameuse{fnum@#1}}{\ignorespaces #3}%
409 %         \@ifnextchar[{\float@ccon}{\egroup}}%
410 %         \long\def\float@ccon[#1]{#1\par\egroup}}%

411 \caption@CheckCommand\@caption{%
412 % hyperref.sty [2007/02/27 v6.75t Hypertext links for LaTeX]
413 \long\def\@caption#1[#2]#3{%
414     \hyper@makecurrent{\@cuptype}%
415     \def\@currentlabelname{#2}%
416     \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
417         \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
418     }%
419     \begingroup
420         \@parboxrestore
421         \if@minipage
422             \@setminipage
423         \fi
424         \normalsize
425         \@makecaption{\csname fnum@#1\endcsname}{%
426             \ignorespaces
427             \ifHy@nesting
428                 \hyper@@anchor{\@currentHref}{#3}%
429             \else
430                 \Hy@raisedlink{\hyper@@anchor{\@currentHref}{\relax}}#3%
431             \fi
432         }%
433     \par
434 \endgroup
435 }}%

436 \caption@CheckCommand\@caption{%
437 % hyperref.sty [2007/04/09 v6.76a Hypertext links for LaTeX]
438 % hyperref.sty [2007/06/12 v6.76h Hypertext links for LaTeX]
439 % hyperref.sty [2007/08/05 v6.76j Hypertext links for LaTeX]
440 \long\def\@caption#1[#2]#3{%
441     \expandafter\ifx\csname if@capstart\expandafter\endcsname
442         \csname iftrue\endcsname
443     \global\let\@currentHref\hc@currentHref
444 \else
445     \hyper@makecurrent{\@cuptype}%
446 \fi
447 \def\@currentlabelname{#2}%
448 \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
449     \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
450 }%
451 \begingroup
452     \@parboxrestore
453     \if@minipage
454         \@setminipage
455     \fi
456     \normalsize
457     \expandafter\ifx\csname if@capstart\expandafter\endcsname
458         \csname iftrue\endcsname
459     \global\@capstartfalse
460     \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%

```

```

461         \else
462         \@makecaption{\csname fnum@#1\endcsname}{%
463         \ignorespaces
464         \ifHy@nesting
465         \hyper@@anchor{\@currentHref}{#3}%
466         \else
467         \Hy@raisedlink{\hyper@@anchor{\@currentHref}{\relax}}#3%
468         \fi
469         }%
470     \fi
471     \par
472     \endgroup
473 } }%

474 \caption@CheckCommand\@caption{%
475 % hyperref.sty [2009/11/27 v6.79k Hypertext links for LaTeX]
476 \long\def\@caption#1[#2]#3{%
477     \expandafter\ifx\csname if@capstart\expandafter\endcsname
478     \csname iftrue\endcsname
479     \global\let\@currentHref\hc@currentHref
480     \else
481     \hyper@makecurrent{\@capttype}%
482     \fi
483     \def\@currentlabelname{#2}%
484     \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
485     \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
486     }%
487     \begingroup
488     \@parboxrestore
489     \if@minipage
490     \@setminipage
491     \fi
492     \normalsize
493     \expandafter\ifx\csname if@capstart\expandafter\endcsname
494     \csname iftrue\endcsname
495     \global\@capstartfalse
496     \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
497     \else
498     \@makecaption{\csname fnum@#1\endcsname}{%
499     \ignorespaces
500     \ifHy@nesting
501     \expandafter\hyper@@anchor\expandafter{\@currentHref}{#3}%
502     \else
503     \Hy@raisedlink{%
504     \expandafter\hyper@@anchor\expandafter{\@currentHref}{\relax}%
505     }%
506     #3%
507     \fi
508     }%
509     \fi
510     \par
511     \endgroup
512 } }%

513 \caption@CheckCommand\@caption{%

```

```

514 % hyperref.sty [2009/12/09 v6.79m Hypertext links for LaTeX]
515 % hyperref.sty [2009/12/28 v6.79z Hypertext links for LaTeX]
516 \long\def\@caption#1[#2]#3{%
517   \expandafter\ifx\csname if@capstart\expandafter\endcsname
518     \csname iftrue\endcsname
519     \global\let\@currentHref\hc@currentHref
520   \else
521     \hyper@makecurrent{\@captype}%
522   \fi
523   \@ifundefined{NR@gettitle}{%
524     \def\@currentlabelname{#2}%
525   }{%
526     \NR@gettitle{#2}%
527   }%
528   \par\addcontentsline{\csname ext@#1\endcsname}{#1}{%
529     \protect\numberline{\csname the#1\endcsname}{\ignorespaces #2}%
530   }%
531   \begingroup
532     \@parboxrestore
533     \if@minipage
534       \setminipage
535     \fi
536     \normalsize
537     \expandafter\ifx\csname if@capstart\expandafter\endcsname
538       \csname iftrue\endcsname
539       \global\@capstartfalse
540       \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces#3}%
541     \else
542       \@makecaption{\csname fnum@#1\endcsname}{%
543         \ignorespaces
544         \ifHy@nesting
545           \expandafter\hyper@@anchor\expandafter{\@currentHref}{#3}%
546         \else
547           \Hy@raisedlink{%
548             \expandafter\hyper@@anchor\expandafter{%
549               \@currentHref
550             }{\relax}%
551           }%
552           #3%
553         \fi
554       }%
555     \fi
556   \par
557 \endgroup
558 }%

559 \caption@CheckCommand\@caption{%
560 % nameref.sty [2006/12/27 v2.28 Cross-referencing by name of section]
561 \long\def\@caption#1[#2]{%
562   \def\@currentlabelname{#2}%
563   \NR@@caption{#1}[{#2}]%
564 }%

565 \caption@CheckCommand\@caption{%
566 % nameref.sty [2009/11/27 v2.32 Cross-referencing by name of section]

```



```

567 \long\def\@caption#1[#2]{%
568 \NR@getttitle{#2}%
569 \NR@caption{#1}[{#2}]%
570 }%
571 \caption@CheckCommand\@caption{%
572 % subfigure.sty [2002/07/30 v2.1.4 subfigure package]
573 \long\def\@caption#1[#2]#3{%
574 \ifundefined{if#1topcap}%
575 {\subfig@oldcaption{#1}[{#2}]{#3}}%
576 {\@nameuse{if#1topcap}%
577 \@listsubcaptions{#1}%
578 \subfig@oldcaption{#1}[{#2}]{#3}%
579 \else
580 \subfig@oldcaption{#1}[{#2}]{#3}%
581 \@listsubcaptions{#1}%
582 \fi}}}%
583 \caption@CheckCommand\@caption{%
584 % subfig.sty [2005/06/28 ver: 1.3 subfig package]
585 \def\@caption{\caption@}%
586 % \long\def\@caption#1[#2]#3{%
587 % \@ifundefined{caption@setfloattype}%
588 % \caption@settype
589 % \caption@setfloattype
590 % \@capttype
591 % \sf@ifpositiontop{%
592 % \@listsubcaptions{#1}%
593 % \sf@old@caption{#1}[{#2}]{#3}%
594 % }%
595 % \sf@old@caption{#1}[{#2}]{#3}%
596 % \@listsubcaptions{#1}%
597 % }%
598 %}%
599 \caption@ifCheckCommand{}{%
600 \caption@InfoNoLine{%
601 Incompatible package detected (regarding \string\@caption).\MessageBreak
602 \string\@caption\space=\space\meaning\@caption}%
603 \caption@setbool{incompatible}{1}}%

```

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

```

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

```

`\ContinuedFloat` is not supported in compatibility mode.

```

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

```

```

\caption@start is not supported in compatibility mode.
616 \caption@AtBeginDocument*{%
617 \let\caption@start\relax
618 \caption@ifundefined\caption@ORI@capstart{}\{%
619 \caption@Debug{%
620 Restore hypcap definition of \string\capstart\@gobble}%
621 \let\capstart\caption@ORI@capstart}%
622 \caption@ifundefined\caption@ORI@float@makebox{}\{%
623 \caption@Debug{%
624 Restore hyperref redefinition of \string\float@makebox\@gobble}%
625 \let\float@makebox\caption@ORI@float@makebox}%
626 }%

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

628 }{%
629 \caption@ifincompatible{%
630 \caption@WarningNoLine{%
631 Forced redefinition of \noexpand\caption since the\MessageBreak
632 unsupported(!) package option 'compatibility=false'\MessageBreak
633 was given}%
634 }{}%

\caption
\@caption 635 \renewcommand*\caption@redefine{%
636 \let\caption\caption@caption
637 \let\@caption\caption@@caption}%
638 \caption@redefine

639 }%
640 \caption@AtBeginDocument*{%
641 \let\caption@ORI@capstart\@undefined
642 \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.
643 \let\caption@ORI@xfloat\@xfloat
644 \def\@xfloat#1[#2]{%
645 \caption@ORI@xfloat{#1}[#2]%
646 \caption@settype{#1}}%

647 }

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.
648 \caption@AtBeginDocument{\caption@redefine}

\@makecaption
649 \let\@makecaption\caption@makecaption

```

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

```

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

```

*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.

```

656 \newcommand\setcaptiontype{%
657   \caption@boxrestore@mini
658   \caption@settype}

```

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

```

659 \newcommand\setcaptionsubtype{%
660   \caption@iftype
661     \caption@setsubtype
662     {\caption@Error{\noexpand\setcaptionsubtype outside float}}}%
663 \newcommand\caption@setsubtype{%
664   \@ifstar
665     {\caption@@settype{sub}*{sub\@capytype}}%
666     {\caption@@settype{sub}{sub\@capytype}}}%

```

`\caption@settype` `\caption@settype*{<type>}`  
 sets `\@capytype` 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 `hycap=true` and the `hycap` package is not loaded).

```

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

```

```

679 \edef\caption@tempa{#3}%
680 \expandafter\ifx\csname @#1capttype\endcsname\caption@tempa \else
681   \ifcaptionsetup@star\else\@nameuse{caption@#1type@warning}\fi
682 \fi
683 \expandafter\let\csname @#1capttype\endcsname\caption@tempa
684 \@nameuse{caption@#1typehook}%
685 \caption@setoptions{#3}%
686 \ifx\caption@opt\relax
687   \@nameundef{caption@#1type@warning}%
688 \else
689   \@namedef{caption@#1type@warning}{\caption@Warning{%
690     The #1caption type was already set to
691     '\csname @#1capttype\endcsname'\MessageBreak}}%
692 \fi
693 \let\caption@ifrefstepcounter\@secondoftwo
694 #2{}{%
695   \let\@currentlabel\caption@undefinedlabel
696 %   \let\@currentHlabel\@undefined
697   \ifx\caption@x@label\@undefined
698     \let\caption@x@label\label
699     \let\label\caption@xlabel
700   \fi
701   \caption@start}}

```

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

```
702 \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.

```
703 \newcommand*\caption@iftype{%
704   \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)

```

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

```

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

```

722 \newcommand*\caption@undefinedlabel{%
723   \protect\caption@xref{\caption@labelname}{\on@line}}
724 \DeclareRobustCommand*\caption@xref[2]{%
725   \caption@WarningNoLine{\noexpand\label without proper \string\caption#2}%
726   \@setref\relax\@undefined{#1}}
727 \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.

```

728 \newcommand*\caption@xlabel{%
729   \caption@withoptargs\caption@@xlabel}
730 \newcommand*\caption@@xlabel[2]{%
731   \caption@@@xlabel
732   \def\caption@labelname{#2}%
733   \caption@x@label#1{#2}}
734 \newcommand*\caption@@@xlabel{%
735   \global\let\caption@@@xlabel\@empty
736   \@bsphack
737   \protected@write\@auxout{}%
738     {\string\providecommand*\string\caption@xref[2]{%
739       \string\@setref\string\relax\string\@undefined{\string##1}}}%
740   \@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)

```

741 \caption@AtBeginDocument{%
742   \def\captionof{\caption@teststar\caption@of{\caption*}\caption}}
743 \newcommand*\caption@of[2]{\setcaptiontype*{#2}#1}

```

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

```

744 \newcommand*\captionlistentry{%
745   \caption@teststar\@captionlistentry\@firstoftwo\@secondoftwo}
746 \newcommand*\@captionlistentry[1]{%
747   \@testopt{\caption@listentry{#1}}\@capttype}
748 \def\caption@listentry#1[#2]#3{%
749   \@bsphack
750   #1{\caption@getttitle{#3}}%
751   {\caption@refstepcounter{#2}}%
752   \caption@makecurrent{#2}{#3}}%
753   \caption@addcontentsline{#2}{#3}%
754   \@esphack}

```

## 11 \captionbox

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

```

755 \newcommand*\captionbox{%
756   \let\captionbox@settype\@gobble
757   \caption@withoptargs\caption@box}

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

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

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

785 \newcommand*\captionbox@hj@default{c}
786 \newcommand*\captionbox@hrule{\hrule\@height\z@}

787 \providecommand*\caption@hj@c{\centering}
788 \providecommand*\caption@hj@l{\raggedright}
789 \providecommand*\caption@hj@r{\raggedleft}
790 \providecommand*\caption@hj@s{}

```

## 12 \ContinuedFloat

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

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

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

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

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

will be used instead of

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

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

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

```

791 \def\ContinuedFloat{%
792   \@ifnextchar[\@Continued@Float\@ContinuedFloat}
793 \def\@Continued@Float[#1]{\addtocounter{#1}{m@ne}
794 \def\@ContinuedFloat{%
795   \caption@iftype
796     {\addtocounter{\@capttype}{m@ne}
797     \caption@ContinuedFloat{\@capttype}%
798     {\caption@Error{\noexpand\ContinuedFloat outside float}}}}
799 \def\caption@ContinuedFloat#1{%
800   \@ifstar{\caption@Continued@Float@{#1}}{\caption@Continued@Float{#1}}}
801 \def\caption@Continued@Float@{%
802   \addtocounter{\@capttype}{@ne}
803   \@stpelt{ContinuedFloat}\stepcounter{ContinuedFloat}%
804   \def\caption@resetContinuedFloat##1{\xdef\caption@CFtype{##1}}%
805   \caption@@ContinuedFloat}
806 \def\caption@Continued@Float#1{%
807   \edef\caption@tempa{#1}%
808   \ifx\caption@tempa\caption@CFtype
809     \stepcounter{ContinuedFloat}%
810     \let\caption@resetContinuedFloat\@gobble
811     \caption@@ContinuedFloat{#1}%
812     \sf@ContinuedFloat{#1}%
813   \else
814     \caption@Error{Continued ‘#1’ after ‘\caption@CFtype’}%
815   \fi}
816 \def\caption@@ContinuedFloat#1{%
817   \expandafter\l@addto@macro\csname the#1\endcsname\theContinuedFloat
818   \@ifundefined{theH#1}{}{%
819     \expandafter\l@addto@macro\csname theH#1\endcsname{%
820       \@alpha@c@ContinuedFloat}}%
821   \caption@setoptions{ContinuedFloat}%
822   \caption@setoptions{continued#1}}
823 \providecommand*\sf@ContinuedFloat[1]{}
824 \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>825 \newcounter{ContinuedFloat} 826 \let\theContinuedFloat\@empty</pre>
<code>\caption@resetContinuedFloat</code>	<pre>\caption@resetContinuedFloat{&lt;type&gt;} If a continuation counter is defined, we reset it. (This one will be called inside \@caption.) 827 \newcommand*\caption@resetContinuedFloat[1]{% 828   \@stpelt{ContinuedFloat}\xdef\caption@CFtype{#1}}</pre>
<code>\phantomcaption</code>	<pre>\phantomcaption Use this one for figures with subcaptions but without main caption. 829 \newcommand\phantomcaption{% 830   \caption@iftype 831     {\caption@refstepcounter\@captype}% 832     {\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>833 \newcommand*\caption@refstepcounter[1]{% 834   \@ifundefined{c@#1}% 835     {\caption@Error{No float type ‘#1’ defined}}% 836     {\caption@resetContinuedFloat{#1}% 837       \caption@@refstepcounter{#1}% 838       \let\caption@ifrefstepcounter\@firstoftwo}} 839 \newcommand*\caption@@refstepcounter{\refstepcounter} 840 \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>841 \caption@ifundefined\kernel@ifnextchar 842   {\newcommand\caption@dblarg[1]{\@ifnextchar[{\#1}{\caption@xdblarg{#1}}}% 843     {\newcommand\caption@dblarg[1]{\kernel@ifnextchar[{\#1}{\caption@xdblarg{#1}}}% 844     \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{&lt;type&gt;}</code> performs these tasks:</p> <ol style="list-style-type: none"> <li>1. Start a new group.</li> <li>2. Define <code>\fnum{&lt;type&gt;}</code> if the caption label format is set to non-default.</li> <li>3. Override the <code>position=</code> setting, if necessary. (for example if set to <code>auto</code> or used inside a <code>supertabular</code>)</li> </ol> <pre>845 \newcommand*\caption@begin[1]{% 846   \begingroup 847     \caption@setfnum{#1}% 848     \caption@fixposition 849     \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>`.

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

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

```
855 \newcommand*\caption@end{%
856   \endgroup
857   \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.

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

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

```

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

```

`\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`.

```

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

```

883 \newcommand*\caption@normalsize{%
884 \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.

```
885 \let\caption@setfloatcapt\@firstofone
```

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

```

\caption@makeanchor 886 \newcommand*\caption@makecurrent[2]{}
\caption@start      887 \let\caption@makeanchor\@firstofone
\caption@@start     888 \let\caption@start\relax
\caption@freezeHref 889 \let\caption@@start\relax
\caption@defrostHref 890 \let\caption@freezeHref\relax
                    891 \let\caption@defrostHref\relax

```

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

```

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

```

## 14 Support for sub-captions

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

```

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

```

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

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

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

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

\caption@subtyphook 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.)

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

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

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

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

```

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

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

```

950 \newcommand*\caption@subphantom{%
951   \caption@checkgrouplevel{sub}\phantomsubcaption
952   \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.

```

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

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

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

```

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

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

```

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

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

```

982 \newcommand*\caption@clearsubcontentslines{%
983   \global\let\caption@subcontentslines\@empty}
```

```

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

```

## 15 Document class & Babel package support

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

```

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

```

### 15.2 The beamer class

```

990 \@ifclassloaded{beamer}{%
991   \caption@InfoNoLine{beamer document class}%

```

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

```

\table 992   \expandafter\let\expandafter\caption@ORI@figure
          \csname\string\figure\endcsname
993   \@namedef{\string\figure}[#1]{%
994     \caption@ORI@figure[#1]%
995     \caption@settype{figure}}
996
997   \expandafter\let\expandafter\caption@ORI@table
998     \csname\string\table\endcsname
999   \@namedef{\string\table}[#1]{%
1000     \caption@ORI@table[#1]%
1001     \caption@settype{table}}
1002 }{}

```

### 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.

```

1003 \caption@ifundefined\scr@caption{}{%
1004   \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)

```

1005 \@nameuse{caption@frenchb}\@nameundef{caption@frenchb}

```

### 15.5 The frenchle/pro package

```

1006 \caption@AtBeginDocument{\caption@ifundefined\frenchTeXmods{}{%
1007   \caption@InfoNoLine{frenchle/pro package is loaded}%
1008   \let\captionfont@ORI\captionfont
1009   \let\captionlabelfont@ORI\captionlabelfont
1010   \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`.

```

1011 \caption@ifundefined\GOfrench
1012   {\let\caption@tempa\@firstofone}%
1013   {\def\caption@tempa{\g@addto@macro\GOfrench}}%
1014 \caption@tempa{%
1015   \let\captionfont\captionfont@ORI
1016   \let\captionfont@ORI\@undefined
1017   \let\captionlabelfont\captionlabelfont@ORI
1018   \let\captionlabelfont@ORI\@undefined
1019   \let\@makecaption\@makecaption@ORI
1020   \let\@makecaption@ORI\@undefined

```

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

```

1021 \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.

```

1022 \let\caption@tcORI\@tablescaption
1023 \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.

```

1024 \let\@eatDP\@undefined
1025 \let\caption@tempa\@empty
1026 \ifx\f@ffrench\fnum@figure
1027   \l@addto@macro\caption@tempa{\let\fnum@figure\f@ffrench}%
1028 \fi
1029 \ifx\f@tfrench\fnum@table
1030   \l@addto@macro\caption@tempa{\let\fnum@table\f@tfrench}%
1031 \fi
1032 \def\f@ffrench{\ifx\listoffigures\relax\else\figurename~\thefigure\fi}%
1033 \def\f@tfrench{\ifx\listoftables\relax\else\tablename~\thetable\fi}%
1034 \caption@tempa
1035 }%
1036 }

```

## 15.6 The hungarian and magyar Babel option

```

1037 \def\caption@tempa#1{%
1038   \@ifundefined{extras#1}\caption@AtBeginDocument\@firstofone{%
1039     \@ifundefined{extras#1}}}%
1040   \caption@InfoNoLine{#1 babel option is loaded}%
1041   \expandafter\addto\csname extras#1\endcsname{%
1042     % reverse changes made by magyar.ldf
1043     \let\@makecaption\caption@makecaption
1044     \babel@save\@makecaption
1045     \caption@redefine

```

```

1046          \babel@save\@caption}%
1047      }}}
1048 \caption@tempa{hungarian}%
1049 \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.
1050 \newcommand\caption@ifpackageloaded[1]{%
1051   \@testopt{\caption@ifpackageloaded{#1}}{}}
1052 \@onlypreamble\caption@ifpackageloaded
1053 \long\def\caption@ifpackageloaded#1[#2]#3#4{%
1054   \@ifpackageloaded{#1}\@firstofone{%
1055     \caption@Debug{#1 package is not loaded (yet)\@gobble}%
1056     \caption@AtBeginDocument}{%
1057       \caption@ifpackageloaded{#1}[#2]{#3}{#4}}}
1058 \@onlypreamble\caption@ifpackageloaded
1059 \newcommand\caption@ifpackageloaded[1]{%
1060   \@testopt{\caption@ifpackageloaded{#1}}{}}
1061 \@onlypreamble\caption@ifpackageloaded
1062 \long\def\caption@ifpackageloaded#1[#2]{%
1063   \@ifpackageloaded{#1}{%
1064     \caption@InfoNoLine{#1 package is loaded}%
1065     \@ifpackagelater{#1}{#2}\@firstoftwo{%
1066       \caption@Error{%
1067         For a successful cooperation we need at least version\MessageBreak
1068         `#2' of package #1,\MessageBreak
1069         but only version\MessageBreak
1070         `\'csname ver@#1.\@pkgextension\endcsname'\MessageBreak
1071         is available}%
1072       \@secondoftwo}%
1073     }{\@secondoftwo}}
1074 \@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.
1075 \newcommand*\caption@clearmargin{%
1076   \setcaptionmargin\z@
1077   \let\caption@minmargin\undefined}

1078 \caption@setbool{needfreeze}{0}
1079 \caption@AtBeginDocument*{%
1080   \caption@ifneedfreeze{%

\caption@freeze \caption@freeze
Used by the fltpage & sidecap package support.
1081 \newcommand*\caption@freeze{%

```

```

1082 \let\caption@frozen@ContinuedFloat\ContinuedFloat
1083 \def\ContinuedFloat{%
1084   \caption@withoptargs\caption@SC@ContinuedFloat}%
1085 \def\caption@SC@ContinuedFloat##1{%
1086   \caption@@freeze{\ContinuedFloat##1}%
1087   \let\caption@frozen@setcounter\setcounter
1088   \let\caption@frozen@addtocounter\addtocounter
1089   \def\setcounter####1####2{\csname c@####1\endcsname####2\relax}%
1090   \def\addtocounter####1####2{\advance\csname c@####1\endcsname ####2\relax}%
1091   \caption@frozen@ContinuedFloat##1%
1092   \let\setcounter\caption@frozen@setcounter
1093   \let\addtocounter\caption@frozen@addtocounter}%
1094 \let\caption@frozen@setup\caption@setup
1095 \def\caption@setup##1{%
1096   \caption@@freeze{\caption@setup{##1}}%
1097   \caption@frozen@setup{##1}}%
1098 \let\caption@frozen@caption\caption
1099 \def\caption{%
1100   \def\caption{%
1101     \caption@Error{%
1102       Only one \noexpand\caption can be placed in this environment}%
1103     \caption@gobble}%
1104   \@ifstar
1105   {\caption@SC@caption*}%
1106   {\let\caption@frozen@refstepcounter\caption@@refstepcounter
1107     \let\caption@@refstepcounter\caption@l@stepcounter
1108     \caption@refstepcounter\@capttype
1109     \let\caption@@refstepcounter\caption@frozen@refstepcounter
1110     \let\@currentlabel\caption@SCLabel
1111     \caption@withoptargs\caption@SC@caption}}%
1112 \long\def\caption@SC@caption##1##2{%
1113   \caption@@freeze{\caption##1{##2}}%
1114   \ignorespaces}%
1115 \let\caption@frozen@label\label
1116 \def\label{%
1117   \caption@withoptargs\caption@SC@label}%
1118 \def\caption@SC@label##1##2{%
1119   \ifx\@currentlabel\caption@SCLabel
1120     \@bsphack
1121     \caption@freeze@label{##1}{##2}%
1122     \@esphack
1123   \else
1124     \caption@frozen@label##1{##2}%
1125   \fi}%
1126 \def\caption@SCLabel{\caption@undefinedlabel}%
1127 \def\caption@freeze@label##1##2{%
1128   \caption@@freeze{\label##1{##2}}}%
1129 \global\let\caption@frozen@content\@empty
1130 \long\def\caption@@freeze{%
1131   \g@addto@macro\caption@frozen@content}%
1132 \def\caption@warmup{%
1133   \let\ContinuedFloat\caption@frozen@ContinuedFloat

```



```

1134      \let\caption@setup\caption@frozen@setup
1135      \let\caption\caption@frozen@caption
1136      \let\label\caption@frozen@label}}%

\caption@defrost \caption@defrost
1137  \newcommand*\caption@defrost{%
1138    \ifx\caption@frozen@caption\@undefined
1139      \caption@frozen@content
1140    \else
1141      \caption@Error{Internal Error:\MessageBreak
1142        \noexpand\caption@defrost in same group as \string\caption@freeze}%
1143    \fi}%

1144  }{}%
1145  \caption@undefbool{needfreeze}}

\caption@warmup \caption@warmup
1146 \let\caption@warmup\relax

```

## 16.1 The float package

The float package usually do not use the L<sup>A</sup>T<sub>E</sub>X 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.)

```

1147 \caption@ifPackageLoaded{float}[2001/11/08 v1.3d]{%
1148   \@ifpackageloaded{floatrow}{%
1149     \caption@ifpackageloaded{floatrow}[2007/08/24 v0.2a]{}{}%
1150   }{}%

```

`\@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>]`.

```

1151 \let\caption@ORI@float@setevery\@float@setevery
1152 \def\@float@setevery#1{%
1153   \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.

```
1154 \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.

```
1155 \renewcommand\caption@setfloatcapt[1]{%
1156 \let\@makecaption\caption@make
1157 \global\setbox\@floatcapt\vbox{%
1158 \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.

```
1159 \float@getstyle\float@style{#1}%
1160 \caption@setstyle*\float@style
1161 \caption@setoptions\float@style
1162 }{}%
1163 \caption@freezeHref % will be defrosted in \float@makebox
1164 \caption@ORI@float@setevery{#1}}%
```

`\caption@typehook` L<sup>A</sup>T<sub>E</sub>X 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.

```
1165 \g@addto@macro\caption@typehook{%
1166 \expandafter\ifx\csname #1name\endcsname\relax
1167 \expandafter\let\csname #1name\expandafter\endcsname
1168 \csname fname@#1\endcsname
1169 \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.

```
1170 \g@addto@macro\fs@plaintop{\def\@fs@mid{\vspace\abovecaptionskip\relax}}%
1171 \g@addto@macro\fs@boxed{\def\@fs@mid{\kern\abovecaptionskip\relax}}%
```

`\float@ifstyle` `\float@ifstyle{⟨type⟩}{⟨if-clause⟩}{⟨else-clause⟩}`

Checks if the given `⟨type⟩` (e.g. figure) is associated with a float style (e.g. boxed).

```
1172 \providecommand*\float@ifstyle[1]{%
1173 \expandafter\ifx\csname fst@#1\endcsname\relax
1174 \expandafter\@secondoftwo
1175 \else
1176 \expandafter\@firstoftwo
1177 \fi}%
```

`\float@getstyle` `\float@getstyle{<cmd>}{<type>}`

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

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

`\fst@Program` will be defined as

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

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

*TODO:* We need to convert the catcodes here.

```
1178 \providecommand*\float@getstyle[2]{%
1179   \edef#1{%
1180     \noexpand\expandafter\noexpand\@gobblefour\noexpand\string
1181     \expandafter\expandafter\expandafter\noexpand
1182     \csname fst@#2\endcsname}%
1183   \edef#1{#1}%
1184   \caption@Debug{floatstyle{#2} = '#1'}}%
```

`\float@setstyle` `\float@setstyle{<type>}{<style>}`

Sets or changes the float style associated with `<type>`.

```
1185 \providecommand*\float@setstyle[2]{%
1186   \expandafter\edef\csname fst@#1\endcsname{%
1187     \expandafter\noexpand\csname fs@#2\endcsname}}%
```

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

```
1188 \providecommand*\float@dostyle[1]{%
1189   \@nameuse{fst@#1}\@float@setevery{#1}}%
```

`\float@ifcaption` `\float@ifcaption{<type>}{<if-clause>}{<else-clause>}`

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

```
1190 \providecommand*\float@ifcaption[1]{%
1191   \expandafter\ifx\csname @float@c@#1\endcsname\float@caption
1192   \expandafter\@firstoftwo
1193   \else
1194   \expandafter\@secondoftwo
1195   \fi}%

1196 }{%
1197   \providecommand*\float@ifstyle[1]{\@secondoftwo}%
1198   \providecommand*\float@ifcaption[1]{\@secondoftwo}%
1199 % \clearcaptionsetup{boxed}% used by the floatrow package?
1200 }
```

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

```
1201 \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’.

```
1202 \caption@ifbool{ruled}{%
1203   \captionsetup[ruled]{margin=0pt,minmargin=0,slc=0}%
1204 }{%
1205   \DeclareCaptionStyle{ruled}{labelfont=bf,labelsep=space,strut=0}%
1206 }
1207 \caption@undefbool{ruled}
```

## 16.2 The floatflt package

```
1208 \caption@ifPackageLoaded{floatflt}[1996/02/27 v1.3]{%
```

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

```
1209 \let\caption@ORI@floatingfigure\floatingfigure
1210 \def\floatingfigure{%
1211   \caption@floatflt{figure}%
1212   \caption@ORI@floatingfigure}%

```

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

```
1213 \let\caption@ORI@floatingtable\floatingtable
1214 \def\floatingtable{%
1215   \caption@floatflt{table}%
1216 %   \caption@setautoposition b%
1217   \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.

```
1218 \newcommand*\caption@floatflt[1]{%
1219   \caption@settype{#1}%
1220   \caption@clearmargin
1221   \caption@setfullparboxrestore
1222   \caption@setoptions{floating#1}}%
1223 }{}
```

## 16.3 The fltpage package

```
1224 \caption@ifPackageLoaded{fltpage}[1998/10/29 v.0.3]{%
1225   \caption@setbool{needfreeze}{1}%

```

\FP@positionLabel **Original code:**

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

1226 \renewcommand\FP@positionLabel{%
1227   FP\FP@capttype-\number\value{FP@\FP@capttype C}-pos}%
```

\FP@helpNote **Original code:**

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

1228 \renewcommand\FP@helpNote[2]{%
1229   \begingroup % save \caption@thepage
1230     \caption@pageref{#2}%
1231     \typeout{FP#1 is inserted on page \caption@thepage!}%
1232   \endgroup}%
```

\FP@floatBegin **Original code:**

```
\newcommand{\FP@floatBegin}[1]{%
  \gdef\@capttype{#1}%
  \global\let\FP@savedCaptionCommand\caption%
  \global\let\FP@savedLabelCommand\label%
  \ifthenelse{\equal{\@capttype}{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}%
}%

1233 \renewcommand*\FP@floatBegin[1]{%
1234   \def\FP@capttype{#1}%
1235   \begin{lrbox}{\FP@floatCorpusBOX}%
1236     \caption@settype*{#1}%
1237     \caption@freeze
1238     \global\let\FP@Label\@empty
1239     \caption@ifFPrefcap
1240     {}%
1241     {\def\caption@freeze@label##1##2{%
1242       \g@addto@macro\FP@Label{\FP@label##1{##2}}}%
1243     \ignorespaces}%
```

\FP@floatEnd **Original code:**

```
\newcommand{\FP@floatEnd}{%
  \end{lrbox}%
  \global\setbox\FP@floatCorpusBOX=\box\FP@floatCorpusBOX
```

```

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

1244 \renewcommand*\FP@floatEnd{%
1245   \end{lrbox}%

1246   \stepcounter{FP@\FP@capttype C}%
1247   \caption@label\FP@positionLabel
1248   \FP@helpNote\FP@capttype\FP@positionLabel

1249   \edef\FP@RestoreCounter{%
1250     \noexpand\setcounter{\FP@capttype}{\the\value\FP@capttype}%
1251     \noexpand\setcounter{ContinuedFloat}{\the\value{ContinuedFloat}}}%

1252   \FP@float
1253     {\FP@positionLabel}% location label test
1254     {\begin\FP@capttype[p!]}%
1255       \usebox\FP@floatCorpusBOX
1256       \let\caption@SClentry\@empty
1257       \def\caption{\caption@dblarg{\@caption\@capttype}}%
1258       \long\def\@caption##1[##2]##3{\def\caption@SClentry{##2}}%
1259       \let\FP@label\label
1260       \let\label\caption@gobble
1261       \caption@defrost
1262       \caption@ifFPlistcap
1263         {\caption@refstepcounter\@capttype
1264           \expandafter\caption@makecurrent\expandafter\@capttype
1265             \expandafter{\caption@SClentry}}%
1266         {\ifx\caption@SClentry\@empty \else
1267           \expandafter\captionlistentry\expandafter{\caption@SClentry}%
1268           \fi}%

```

```

1269         \caption@makeanchor\relax
1270         \FP@Label
1271     \end\FP@captype}%
1272     {\FP@RestoreCounter
1273     \@ifundefined{theH\FP@captype}{}{%
1274         \expandafter\l@addto@macro\csname theH\FP@captype\endcsname{.FP}}}%
1275     {\begin\FP@captype[b!]}%
1276         \let\FP@savedSetfnumCommand\caption@setfnum
1277         \def\caption@setfnum##1{%
1278             \FP@savedSetfnumCommand{##1}%
1279             \ifx\FP@guide\empty \else
1280                 \expandafter\l@addto@macro\csname fnum@##1\endcsname{\ \FP@guide}}}%
1281             \fi}%
1282         \setlength\abovecaptionskip{2pt plus 2pt minus 1pt}% length above caption
1283         \setlength\belowcaptionskip{2pt plus 2pt minus 1pt}% length below caption
1284         \caption@setoptions{FP\@captype}%
1285         \FP@separatorCaption
1286         \caption@ifFPlistcap{}{\let\caption@addcontentsline\@gobbletwo}%
1287         \caption@defrost
1288     \end\FP@captype}%
1289 }%

1290 \caption@For{typelist}{%
1291     \newcounter{FP@#1C}%
1292     \newenvironment{FP#1}{\FP@floatBegin{#1}}{\FP@floatEnd}}%

1293 }{%
1294     \let\caption@ifFPlistcap\@undefined
1295     \let\caption@ifFPrefcap\@undefined
1296 }

```

## 16.4 The hyperref package

```

1297 \caption@IfPackageLoaded{hyperref}[2003/11/30 v6.74m]{%
1298     % Test if hyperref has stopped early
1299     \caption@ifundefined\IfHyperBoolean{%
1300         \caption@set@bool\caption@ifhyp@stoppeearly0%
1301         \caption@ifundefined\H@refstepcounter
1302             {\caption@set@bool\caption@ifhyp@stoppeearly1}}{%
1303             \caption@ifundefined\hyper@makecurrent
1304             {\caption@set@bool\caption@ifhyp@stoppeearly1}}{%
1305             \caption@ifundefined\measuring@true
1306             {\caption@set@bool\caption@ifhyp@stoppeearly1}}}%
1307     }{%
1308         \def\caption@ifhyp@stoppeearly{\IfHyperBoolean{stoppeearly}}%
1309     }%
1310     \caption@ifhyp@stoppeearly{% hyperref has stopped early
1311         \caption@InfoNoLine{%
1312             Hyperref support is turned off\MessageBreak
1313             because hyperref has stopped early}%
1314     }{%
1315         \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`.

```

1316      \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.

1317      \renewcommand*\caption@makecurrent[2]{%
1318          \caption@makecurrentHref{#1}%
1319          \caption@Debug{hyperref current=\@currentHref}%
1320          \caption@getttitle{#2}}%

1321      \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.

1322      \renewcommand\caption@makeanchor[1]{%
1323          \caption@Debug{hyperref anchor: \@currentHref}%
1324          % If we cannot have nesting, the anchor is empty.
1325          \ifHy@nesting
1326              \expandafter\hyper@@anchor\expandafter{\@currentHref}{#1}%
1327          \else
1328              \Hy@raisedlink{%
1329                  \expandafter\hyper@@anchor\expandafter{\@currentHref}{\relax}%
1330              }#1%
1331          \fi}%

1332      \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.

1333      \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.

1334      \def\caption@start{\caption@ifhypcap\caption@start@relax}%
1335      \def\caption@start@{%

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

1336          \caption@makestart\@captype
1337          \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.)

1338          \global\@capstarttrue
1339          \let\hc@currentHref\@currentHref
1340          \def\caption@makecurrentHref##1{%
1341              \global\@capstartfalse
1342              \global\let\@currentHref\hc@currentHref}%

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

1343          \let\caption@makeanchor\@firstofone

1344      }%

```



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

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

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

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

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

```

1345 \newcommand*\caption@makestart[1]{%
1346 \begingroup
1347 \Hy@hypertextnamesfalse
1348 % \gdef\@currentHlabel{}%
1349 \hyper@makecurrent{#1.caption}%
1350 \endgroup
1351 \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.

```

1352 \newcommand*\caption@startanchor[1]{%
1353 \ifvmode\begingroup
1354 \caption@Debug{hypcap anchor: #1 (vertical mode)}%
1355 \@tempdima\prevdepth
1356 \nointerlineskip
1357 \vspace*{-\caption@hypcapspace}%
1358 \caption@anchor{#1}%
1359 \vspace*{\caption@hypcapspace}%
1360 \prevdepth\@tempdima
1361 \endgroup\else
1362 \caption@Debug{hypcap anchor: #1 (horizontal mode)}%
1363 \caption@anchor{#1}%
1364 \fi}%%
```

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

```

1365 \newcommand*\caption@anchor[1]{%
1366 \ifmeasuring@ \else
1367 \caption@raisedlink{\hyper@anchorstart{#1}\hyper@anchorend}%
1368 \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`.)

```
1369 \ifx\HyperRaiseLinkLength\@tempdima
1370 \def\caption@raisedlink#1{\ifvmode#1\else\Hy@raisedlink{#1}\fi}%
1371 \else
1372 \let\caption@raisedlink\Hy@raisedlink
1373 \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.

```
1374 \def\caption@@start{%
1375 \caption@ifundefined\hc@currentHref{%
1376 \caption@Warning{%
1377 The option ‘hycap=true’ will be ignored for this\MessageBreak
1378 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`.

```
1379 \def\caption@freezeHref{%
1380 \let\caption@ORI@start\caption@start
1381 \def\caption@start{\let\caption@start\caption@ORI@start}%
1382 % \let\caption@ORI@@start\caption@@start
1383 % \l@addto@macro\caption@subtyphook{%
1384 % \let\caption@@start\caption@ORI@@start}%
1385 \global\let\caption@currentHref\@undefined
1386 \def\caption@@start{\global\let\caption@currentHref\@currentHref}%
1387 \let\caption@ORI@setfloatcapt\caption@setfloatcapt
1388 \renewcommand*\caption@setfloatcapt{%
1389 \ifx\caption@currentHref\@undefined \else
1390 \let\caption@makeanchor\@firstofone
1391 \fi
1392 \caption@ORI@setfloatcapt}}%
```

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

```
1393 \def\caption@defrostHref{%
1394 \ifx\caption@currentHref\@undefined \else
1395 \caption@startanchor\caption@currentHref
1396 \global\let\caption@currentHref\@undefined
1397 \fi}%
```

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

```
1398 \caption@ifundefined\HyOrg@float@makebox{ }{%
1399 \caption@Debug{%
1400 Redefining \noexpand\float@makebox (again)\@gobble}%
1401 \let\caption@ORI@float@makebox\float@makebox % save for compatibility mode
1402 \renewcommand\float@makebox[1]{%
1403 \HyOrg@float@makebox{#1}\relax \caption@defrostHref}}%
1404 }%
```

```
1405 }{} }
```

## 16.5 The hypcap package

```
1406 \caption@ifpackageloaded{hypcap}{% v1.0
1407 \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.

```
1408 \let\caption@ORI@capstart\capstart % save for compatibility mode
1409 \caption@ifundefined\capstarttrue % check for v1.10 of hypcap package
1410 {\def\capstart{\caption@start@}}%
1411 {\def\capstart{\ifcapstart\caption@start@\fi}}%
1412 \let\caption@start\relax
1413 \let\caption@@start\relax
```

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

```
1414 \caption@set@bool\caption@ifhypcap 1%
1415 \renewcommand*\caption@hypcapspace{\hypcapspace}%
1416 \fi}{} }
```

## 16.6 The listings package

```
1417 \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.

```
1418 \let\caption@ORI@lst@MakeCaption\lst@MakeCaption
1419 \def\lst@MakeCaption#1{% #1 is ‘t’ or ‘b’
1420 \begingroup
```

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

```
1421 \ifdim\hsize>\linewidth
1422 \hsize\linewidth
1423 \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`.)

```
1424 \caption@setposition{#1}%
1425 \caption@iftop{%
1426 \@tempdima\belowcaptionskip
1427 \belowcaptionskip\abovecaptionskip
1428 \abovecaptionskip\@tempdima}{}%
```

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

```
1429 \caption@setup{rule=0}%
```

Afterwards we set the local ‘lstlisting’ options.

```
1430 \caption@setoptions{lstlisting}%
```

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

```
1431 \caption@setautoposition{#1}%
```

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

```
1432 \caption@begin{lstlisting}%
```

```
1433 \caption@ORI@lst@MakeCaption{#1}%
```

```
1434 \caption@end
```

```
1435 \endgroup}%
```

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

```
\lst@maketitle 1436 \def\lst@makecaption{\caption@starfalse\@makecaption}%
```

```
1437 \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.

```
1438 \providecommand*\ext@lstlisting{lol}%
```

```
1439 }{ }
```

## 16.7 The longtable package

\LTcapttype \LTcapttype is preset to table.

```
1440 \providecommand*\LTcapttype{table}
```

```
1441 \caption@ifPackageLoaded{longtable}[1995/05/24 v3.14]{%
```

```
1442 \RequirePackage{ltcaption}[2007/09/01]%
```

```
1443 \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.

```
1444 \caption@AtBeginDocument{%
```

```
1445 \let\caption@ORI@LT@array\LT@array
```

```
1446 \renewcommand*\LT@array{%
```

\captionsetup for longtable:

```
1447 \global\let\caption@opt@@longtable\@undefined
```

```
1448 \def\captionsetup{%
```

```
1449 \noalign\bgroup
```

```
1450 \@ifstar\@captionsetup\@captionsetup}% gobble *
```

```
1451 \def\@captionsetup##1{\LT@captionsetup{##1}\egroup}%
```

```
1452 \def\LT@captionsetup##1{%
```

```
1453 \captionsetup@startrue\caption@setup@options[@longtable]{##1}%
```

```
1454 \global\let\caption@opt@@longtable\caption@opt@@longtable}%
```

```

\captionabove & \captionbelow for longtable: (KOMA-Script document class)
1455 \def\@captionabovetrue{\LT@captionsetup{position=t}}%
1456 \def\@captionabovefalse{\LT@captionsetup{position=b}}%
\captionlistentry for longtable:
1457 \def\captionlistentry{%
1458 \noalign\bgroup
1459 \@ifstar{\egroup\LT@captionlistentry}% gobble *
1460 {\egroup\LT@captionlistentry}}%
1461 \def\LT@captionlistentry##1{%
1462 \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
1463 % \caption@ifhyppcap{%
1464 % \let\caption@ORI@hyper@makecurrent\hyper@makecurrent
1465 % \def\hyper@makecurrent##1{%
1466 % \let\hyper@makecurrent\caption@ORI@hyper@makecurrent
1467 % \caption@makestart{##1}%
1468 % \let\Hy@LT@currentHlabel\@currentHlabel
1469 % \let\Hy@LT@currentHref\@currentHref
1470 % \def\hyper@makecurrent###1{%
1471 % \let\@currentHlabel\Hy@LT@currentHlabel
1472 % \let\@currentHref\Hy@LT@currentHref}}%
1473 % \let\caption@ORI@ContinuedFloat\ContinuedFloat
1474 % \def\ContinuedFloat{\noalign{%
1475 % \gdef\caption@setContinuedFloat{%
1476 % \let\caption@resetContinuedFloat\@gobble}%
1477 % \def\caption@setoptions####1{%
1478 % \g@addto@macro\caption@setContinuedFloat{%
1479 % \caption@setoptions{####1}}}%
1480 % \let\@capttype\LTcapttype
1481 % \caption@ORI@ContinuedFloat}}%
1482 % }{%
1483 % \def\ContinuedFloat{\noalign{%
1484 % \caption@Error{%
1485 % \noexpand\ContinuedFloat inside longtables\MessageBreak
1486 % is only available with 'hyppcap=true'}}}%
1487 % }%
1488 % \global\let\caption@setContinuedFloat\@empty
1489 % \def\ContinuedFloat{\noalign{%
1490 % \caption@Error{\noexpand\ContinuedFloat outside float}}}%
1491 % \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}`:

```
1492 \long\def\LT@caption#1[#2]#3{%
1493   \LT@makecaption#1{\csname fnum@\LTcapttype\endcsname}{#3}%
1494   \LT@captionlistentry{#2}}%
```

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

The original definition:

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

Our definition:

```
1495 \renewcommand\LT@makecaption[3]{%
1496   \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`.)

```
1497   \caption@settype*\LTcapttype
1498   \ifdim\LTcapwidth=4in \else
1499     \setcaptionwidth\LTcapwidth
1500   \fi
1501   \caption@setoptions{longtable}%
1502 %   \caption@setContinuedFloat
1503   \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!

```
1504   \caption@setautoposition{\ifcase\LT@rows t\else b\fi}%
```

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

```
1505   \caption@startrue#1\caption@starfalse
1506   \caption@resetContinuedFloat\LTcapttype
1507   \caption@begin\LTcapttype
1508   \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...)

```
1509 \vskip-\ht\strutbox
```

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

```
1510 \caption@iftop{\vskip\belowcaptionskip}{\vskip\abovecaptionskip}%
1511 \caption@@make{#2}{#3}\endgraf
1512 \caption@iftop{\vskip\abovecaptionskip}{\vskip\belowcaptionskip}%
1513 \caption@end}%
```

```
1514 {} }
```

## 16.8 The picinpar package

```
1515 \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.

```
1516 \long\def\figwindow[#1,#2,#3,#4] {%
1517 \caption@window{figure}%
1518 \caption@setoptions{figwindow}%
1519 \begin{window}[#1,#2,{#3},\caption@wincaption{#4}] }%
1520 \long\def\tabwindow[#1,#2,#3,#4] {%
1521 \caption@window{table}%
1522 \caption@setoptions{tabwindow}%
1523 \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.

```
1524 \newcommand*\caption@window[1]{%
1525 \let\@makecaption\caption@@make
1526 \caption@setautoposition b%
1527 \caption@settype{#1}%
1528 \caption@clearmargin
1529 \caption@setfullparboxrestore}%
```

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

```
1530 \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`.

```
1531 \ifdim\picwd=\z@
1532 \let\caption@makecurrent\@gobbletwo
1533 \let\caption@@start\relax
1534 \caption@prepareslc
1535 \fi
```

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

```

1536 \edef\@tempa{\expandafter\noexpand\@car#1\@nil}%
1537 \if\@tempa*%
1538 \let\@tempa\@firstofone
1539 \else\if\@tempa[%]
1540 \let\@tempa\@firstofone
1541 \else
1542 \let\@tempa\@empty
1543 \fi\fi
1544 \expandafter\caption\@tempa{#1}}%

1545 }{}

```

## 16.9 The picins package

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

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

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

```
1546 \newcommand*\piccaptiontype[1]{\def\@piccaptiontype{#1}}
```

```
1547 \caption@ifpackageloaded{picins}{%
```

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

```

1548 \caption@ifundefined\@piccaptiontype{%
1549 \caption@iftype{%
1550 \let\@piccaptiontype\@capytype
1551 }%
1552 \def\@piccaptiontype{figure}%
1553 }%
1554 }{}%
1555 \let\@capytype\@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.

```
1556 \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. `\@capytype` 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.

```

1557 \let\caption@ORI@make@piccaption\make@piccaption
1558 \def\make@piccaption{%
1559   \let\caption@ORI\caption
1560   \long\def\caption[##1]##2{%
1561     \caption@freezeHref % will be defrosted in \ivparpic
1562     \caption@settype\@piccaptiontype
1563 %   \ifnum\c@piccaptionpos>2\relax
1564     \caption@clearmargin
1565 %   \else
1566 %     \captionwidth\z@ % do not use "width=" setting
1567 %   \fi
1568     \caption@setfullparboxrestore
1569     \caption@setoptions{parpic}%
1570     \caption@setautoposition b%
1571     \expandafter\expandafter\expandafter\caption@ORI
1572       \expandafter\expandafter\expandafter[%
1573       \expandafter\expandafter\expandafter{%
1574       \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
1575   \caption@ORI@make@piccaption
1576   \let\caption\caption@ORI}%

```

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

```

1577 \let\caption@ORI@ivparpic\ivparpic
1578 \def\ivparpic(#1,#2)(#3,#4)[#5][#6]#7{%
1579   \let\caption@ORI@noindent\noindent
1580   \def\noindent{%
1581     \caption@defrostHref
1582     \let\noindent\caption@ORI@noindent
1583     \noindent}%
1584   \caption@ORI@ivparpic(#1,#2)(#3,#4)[#5][#6]{#7}%
1585   \let\noindent\caption@ORI@noindent}%

```

```

1586 }{%
1587 \let\piccaptiontype\@undefined
1588 }

```

## 16.10 The rotating package

```

1589 \caption@ifPackageLoaded{rotating}[1995/08/22 v2.10]{%
\rotcaption Make \rotcaption* work.
1590 \def\rotcaption{\let\@makecaption\@makerotcaption\caption}%
1591 % \let\@rotcaption\@undefined

\rotcaptionof Make \rotcaptionof(*) work.
1592 \def\rotcaptionof{%
1593 \caption@teststar\caption@of{\rotcaption*}\rotcaption}%

\@makerotcaption Original (bugfixed) code:

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

```

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

```

1594 \long\def\@makerotcaption#1#2{%
1595 \ifdim\captionwidth=\z@
1596 \setcaptionwidth{.8\textheight}%
1597 \caption@slc{#1}{#2}{.8\vsiz}{%
1598 \let\caption@makerot\caption@@make
1599 \caption@clearmargin
1600 % \long\def\caption@parbox##1##2{\hbox{\hsiz=.8\textheight\relax##2}}%
1601 % (not needed because \rotatebox uses an \hbox anyway)
1602 \let\caption@parbox\@secondoftwo}%
1603 \caption@set@bool\caption@ifslc0% been there, done that
1604 \fi
1605 \rotatebox{90}{\caption@makerot{#1}{#2}}%
1606 \nobreak\hspace{12pt}}%
1607 \newcommand\caption@makerot[2]{%
1608 \begin{minipage}\captionwidth\caption@@make{#1}{#2}\end{minipage}}%
1609 \caption@For{typelist}{%
1610 \newenvironment{sideways#1}{\@rotfloat{#1}}{\end@rotfloat}%
1611 \newenvironment{sideways#1*}{\@rotdblfloat{#1}}{\end@rotdblfloat}}%
1612 {}

```

## 16.11 The sidecap package

```
1613 \caption@ifpackageloaded{sidecap}[2003/06/06 v1.6f]{%
1614   \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.

```
1615   \let\caption@ORI@SC@zfloat\SC@zfloat
1616   \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.

```
1617     \caption@ORI@SC@zfloat{#1}{#2}{#3}[#4]%
1618     \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.

```
1619     \caption@settype*{#2}%
1620     \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.

```
1621     \let\SC@label\label}%
1622   \providecommand*\SC@RestoreCommands{%
1623     \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.

```
1624   \let\caption@ORI@endSC@FLOAT\endSC@FLOAT
1625   \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.

```
1626     \let\caption@ORI@settype\caption@settype
1627     \def\caption@settype##1{% will be done in \@xfloat
1628       \caption@ORI@settype*{##1}% do not change \@currentlabel
1629       \caption@setSC@justify
1630 %%% \caption@setoptions{SCfloat}%
1631       \caption@setoptions{SC@\capttype}%
1632       \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

```
1633     \def\caption@setSC@justify{%
1634       \caption@clearmargin
```

```

1635         \ifx\SC@justify\@empty \else
1636             \let\caption@hj\SC@justify
1637             \let\SC@justify\@empty
1638         \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!

```

1639     \let\caption\SC@orig@caption
1640     \def\SC@orig@caption[##1]##2{\caption@defrost}%

```

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

```

1641     \caption@setSC@justify % for compatibility mode
1642     \caption@ORI@endSC@FLOAT}%

1643 \newcommand*\caption@For@SC[2]{%
1644     \def#1{b}% = \sidecaptionvpos{#2}{b} (v1.6)
1645     \newenvironment{SC#2}%
1646         {\SC@float[#1]{#2}}{\endSC@float}%
1647     \newenvironment{SC#2*}%
1648         {\SC@dblfloat[#1]{#2}}{\endSC@dblfloat}}%
1649 \@onlypreamble\caption@For@SC
1650 \caption@For{typelist}{%
1651     \expandafter\caption@For@SC\csname SC@#1@vpos\endcsname{#1}}}%
1652 }{}

```

## 16.12 The subfigure package

```

1653 \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.

```

1654 \def\sf@ifpositiontop{%
1655     \ifx\@capttype\@undefined
1656         \expandafter\@gobbletwo
1657     \else\ifx\@capttype\relax
1658         \expandafter\expandafter\expandafter\@gobbletwo
1659     \else
1660         \expandafter\expandafter\expandafter\sf@if@position@top
1661     \fi\fi}

1662 \def\sf@if@position@top{%
1663     \@ifundefined{if\@capttype topcap}%
1664         {\@gobbletwo}%
1665         {\@nameuse{if\@capttype topcap}%
1666         \expandafter\@firstoftwo
1667     \else
1668         \expandafter\@secondoftwo
1669     \fi}}

1670 }{}

```

## 16.13 The supertabular and xtab packages

```

1671 \caption@ifpackageloaded{supertabular}[2002/07/19 v4.1e]{%
\tablecaption Make \topcaption* and \bottomcaption* work.
1672   \renewcommand*\tablecaption{%
1673     \caption@star
1674     {\refstepcounter{table}}%
1675     {\caption@dblarg{\@xtablecaption}}}%

\@xtablecaption Make \nameref and \autoref work.
1676   \let\caption@ORI@xtablecaption\@xtablecaption
1677   \long\def\@xtablecaption[#1]#2{%
1678     \caption@getttitle{#2}%
1679     \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

1680   \long\def\ST@caption#1[#2]#3{\par%
1681     \caption@settype*{#1}%
1682     \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.

1683   \def\caption@fixposition{%
1684     \caption@setposition{\if@topcaption t\else b\fi}}%

1685   \caption@beginex{#1}{#2}{#3}%
1686   \caption@boxrestore
1687   \caption@normalsize
1688   \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
1689   \caption@end}%

1690 }{}

1691 \caption@ifpackageloaded{xtab}[2000/04/09 v2.3]{%

\tablecaption Make \topcaption* and \bottomcaption* work.
1692   \renewcommand*\tablecaption{%
1693     \caption@star
1694     {\refstepcounter{table}}%
1695     {\caption@dblarg{\@xtablecaption}}}%

```

```

\@xtablecaption Make \nameref and \autoref work.
1696 \let\caption@ORI@xtablecaption\@xtablecaption
1697 \long\def\@xtablecaption[#1]#2{%
1698 \caption@getttitle{#2}%
1699 \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}}%
\beginngroup
\@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}}

1700 \long\def\ST@caption#1[#2]#3{\par%
1701 \caption@settype*{#1}%
1702 \caption@setoptions{xtabular}%
1703 \def\caption@fixposition{%
1704 \caption@setposition{\if@topcaption t\else b\fi}}%
1705 \@initisotab
1706 \caption@beginex{#1}{#2}{#3}%
1707 \caption@boxrestore
1708 \caption@normalsize
1709 \@makecaption{\csname fnum@#1\endcsname}{\ignorespaces #3}\par
1710 \caption@end
1711 \global\advance\ST@pageleft -\PWSTcapht
1712 \ST@trace\tw@{Added caption. Space left for xtabular: \the\ST@pageleft}}%

1713 {}

```

## 16.14 The threeparttable package

```

1714 \caption@IfPackageLoaded{threeparttable}[2003/06/13 v3.0]{%
\threeparttable Unfortunately \@capytype is not set when \TPT@common will be used, so we have to
redefine \threeparttable and \measuredfigure instead.
1715 \let\caption@ORI@threeparttable\threeparttable
1716 \renewcommand*\threeparttable{%
1717 \caption@settype{table}%
1718 \caption@setposition a% ?
1719 \caption@clearmargin
1720 \caption@setoptions{threeparttable}%
1721 \caption@ORI@threeparttable}%

```

`\measuredfigure` Same here...

```
1722 \let\caption@ORI@measuredfigure\measuredfigure
1723 \renewcommand*\measuredfigure{%
1724   \caption@settype{figure}%
1725   \caption@setposition a% ?
1726   \caption@clearmargin
1727   \caption@setoptions{measuredfigure}%
1728   \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}

1729 \def\TPT@caption#1[#2]#3{%
1730   \gdef\TPT@docapt{%
1731     \global\let\TPT@docapt\@undefined
1732     \caption@setautoposition\caption@TPT@position
1733     \TPT@LA@caption{#1}[{#2}]{#3}}%
1734   \ifx\TPT@hsize\@empty
1735     \let\label\TPT@gatherlabel % Bug: does not work for measuredfigures
1736     \gdef\caption@TPT@position{t}%
1737     \g@addto@macro\TPT@docapt\caption@TPT@eatvskip
1738   \else
1739     \def\caption@TPT@position{b}%
1740     \TPT@docapt
1741   \fi
1742   \ignorespaces}%
1743 %\newcommand*\caption@TPT@eatvskip{\vskip-.2\baselineskip}%
1744 \def\caption@TPT@eatvskip#1\vskip{#1\@tempdima=}%

1745 {} }
```

## 16.15 The wrapfig package

```
1746 \caption@ifPackageLoaded{wrapfig}{% ver 3.3 (Oct 12, 1999)
```

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

```
1747 \providecommand*\float@ifstyle[1]{%
1748   \expandafter\ifx\csname fst@#1\endcsname\relax
1749   \expandafter\@secondoftwo
1750   \else
1751   \expandafter\@firstoftwo
1752   \fi}%
```

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

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

```

1753 \newcommand*\caption@restylewrapfloat[1]{%
1754   \expandafter\ifx\csname caption@OUR@wrap#1\expandafter\endcsname
1755     \csname wrap#1\endcsname
1756   \caption@Error{%
1757     For a successful cooperation of the 'wrapfig' package\MessageBreak
1758     with the 'float' package you should load the 'wrapfig'\MessageBreak
1759     package *after* the 'float' package}%
1760   \else
1761     \expandafter\let\csname caption@ORI@wrap#1\expandafter\endcsname
1762       \csname wrap#1\endcsname
1763     \@namedef{wrap#1}{\caption@wrapfloat{#1}}%
1764     \expandafter\let\csname caption@OUR@wrap#1\expandafter\endcsname
1765       \csname wrap#1\endcsname
1766   \fi}%

```

`\caption@wrapfloat`

```

1767 \newcommand*\caption@wrapfloat[1]{%
1768   \caption@settype*{#1}%
1769   \float@ifstyle{#1}{%
1770     \ifx\WF@floatstyhook\@undefined
1771       \caption@Error{%
1772         For a successful cooperation of the 'wrapfig' package\MessageBreak
1773         with the 'float' package you should use at least\MessageBreak
1774         'wrapfig' version 3.6}%
1775     \else
1776       \float@dostyle{#1}%
1777       \fi}}%
1778   \caption@clearmargin
1779   %% \caption@setoptions{wrapfloat}%
1780   \caption@setoptions{wrap#1}%
1781   \@nameuse{caption@ORI@wrap#1}}%

```

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

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

```

1782 \caption@restylewrapfloat{figure}%
1783 \caption@restylewrapfloat{table}%
1784 \caption@For{typelist}{%
1785   \newenvironment{wrap#1}{\wrapfloat{#1}}{\endwrapfloat}%
1786   \caption@restylewrapfloat{#1}}%
1787 \ifx\WF@floatstyhook\@undefined \else % wrapfig v3.6

```

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

```

1788 \caption@ifundefined\float@restyle{}{%
1789   \toks@=\expandafter{\float@restyle{#1}}% (env may or may not be defined)

```



```

1790         \caption@restylewrapfloat{#1}}%
1791         \edef\@tempa{\def\noexpand\float@restyle##1{\the\toks@}}%
1792         \@tempa}% perform redefinitions

\wrapfloat An additional check of the package load order: If both, neither the wrapfig package
nor the caption package haven't catch \float@restyle, we finally splash down at
\wrapfloat.
1793     \let\caption@ORI@wrapfloat\wrapfloat
1794     \def\wrapfloat#1{%
1795         \float@ifstyle{#1}{%
1796             \caption@Error{%
1797                 For a successful cooperation of the 'wrapfig' package\MessageBreak
1798                 with the 'float' package you should load the 'wrapfig'\MessageBreak
1799                 package *right after* the 'float' package}}}%
1800         \caption@ORI@wrapfloat{#1}}%

1801     \fi                                     % wrapfig v3.6

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

Our code:
1802 \def\WF@rapt[#1]#2{% final two args: #1 = overhang, #2 = width,
1803     \gdef\WF@ovh{#1}% hold overhang for later, when \width is known
1804     \global\setbox\WF@box\top\bggroup \setlength\hsize{#2}%
1805     \caption@start
1806     \ifdim\hsize>\z@ \@parboxrestore \else
1807     \setbox\z@\hbox\bggroup \let\wf@@caption\caption \let\caption\wf@caption
1808     \ignorespaces \fi}%

1809 } {}

```

## References

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

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