

The ltcaption package*

Axel Sommerfeldt
`axel.sommerfeldt@f-m.fm`

2009/03/30

Abstract

This package fixes caption problems with other-than-centered aligned longtables.
(solves [L^AT_EX PR tools/3387](#))

Contents

1	The user interface	2
1.1	Further justification	2
1.2	Bonus features	2
2	Spot the difference	4
3	The Implementation	7
3.1	Identification	7
3.2	User interface	7
3.3	The longtable patch	7
3.4	The longtable* environment	11
3.5	Adaption for KOMA-Script	11

*This package has version number v1.3, last revised 2011/09/12.

1 The user interface

The content of `\caption` in longtables is usually centered to the content of the longtable itself. This is sufficient for centered longtables, but for left or right aligned longtables this results in captions moved into the left or right page margin.

To solve this problem just include this package *after* the longtable package[1], e.g.:

```
\usepackage{longtable, ltcaption}
```

Afterwards the captions of all longtables should be aligned as expected, even for not-centered longtables.

1.1 Further justification

`\LTcapskip` This length is controlling the skip between the caption and the contents below the caption (which is usually the longtable contents if you place the caption above the longtable), and it can be altered with `\setlength\LTcapskip{...}`. When the `ltcaption` package is loaded it will be set to `\abovecaptionskip` which usually represents the skip between caption and contents in floating environments. (Without this package, the longtable package uses `\baselineskip` here.)

`\LTcapleft` You can alter the centering of the caption box (of width `\LTcapwidth`) by setting the
`\LTcapright` lengths `\LTcapleft` & `\LTcapright` to appropriate values. These are set to `\fill` by default, just like the values `\LTleft` & `\LTright`.

`\LTcapmarginfalse` Another option is the usage of the command `\LTcapmarginfalse` which makes the `ltcaption` package using the values `\LTleft` & `\LTright` instead of `\LTcapleft` & `\LTcapright`.

Note: If the `ltcaption` package will be used with one of the NTG document classes[2], `\CaptionLabelFont` & `\CaptionTextFont` will not only be used for figure & table captions, but for longtable captions as well.

Note: These lengths & commands do not work when the `ltcaption` package is used with one of the KOMA-Script classes[3] `scrartcl`, `scrreprt` or `scrbook`, the KOMA-Script settings for captions are used instead. Same with the `caption` package which also uses its own options and settings.

1.2 Bonus features

`\LTcapttype` As a bonus feature this package patches the longtable package so `\LTcapttype` will be used internally instead of the fixed caption type 'table'. So for example this code snippet:

```
\renewcommand\LTcapttype{figure}
\begin{longtable}{ll}
\caption{An example longtable}\\
A & B \\
\end{longtable}
```

will result in a longtable like this:

Figure 7: An example longtable

`longtable*` Another bonus feature is the addition of the environment `longtable*` which does not increment the `table` counter. (As a consequence, `\caption` is not available for such `longtables`, but `\caption*` still is.)

2 Spot the difference

Without the `ltablex` package:

Table 1: Left aligned longtable left aligned longtable left aligned longtable

This is only a test

Table 2: Right aligned longtable right aligned longtable right aligned longtable

This is only a test

Table 3: Centered longtable centered longtable centered longtable centered longtable

This is only a test

With the `ltablex` package (and the default value of `\LTcapwidth`):

Table 4: Left aligned longtable left aligned longtable left aligned longtable

This is only a test

Table 5: Right aligned longtable right aligned longtable right aligned longtable

This is only a test

Table 6: Centered longtable centered longtable centered longtable centered longtable

This is only a test

With the `ltablex` package and `\LTcapwidth=\linewidth`:

Table 7: Left aligned longtable left aligned longtable left aligned longtable left aligned longtable

This is only a test

Table 8: Right aligned longtable right aligned longtable right aligned longtable right aligned longtable

This is only a test

Table 9: Centered longtable centered longtable centered longtable centered longtable

This is only a test

With the `ltablex` package and `\LTcapleft=0pt` resp. `\LTcapright=0pt`:

Table 10: Left aligned longtable left aligned longtable left aligned longtable

This is only a test

Table 11: Right aligned longtable right aligned longtable right aligned longtable

This is only a test

With the `ltablex` package and `\LTcapleft=\tabcolsep` resp. `\LTcapright=\tabcolsep`:

Table 12: Left aligned longtable left aligned longtable left aligned longtable

This is only a test

Table 13: Right aligned longtable right aligned longtable right aligned longtable

This is only a test

With the `lrcaption` package and `\LTcapmarginfalse`:

Table 14: Left aligned longtable left aligned longtable left aligned
longtable

This is only a test

Table 15: Right aligned longtable right aligned longtable right aligned
longtable

This is only a test

Table 16: Centered longtable centered longtable centered longtable cen-
tered longtable

This is only a test

3 The Implementation

3.1 Identification

```
1 \NeedsTeXFormat{LaTeX2e}[1994/12/01]
2 \ProvidesPackage{ltcaption}[2011/09/12 v1.3 longtable captions (AR)]
3 \@ifpackageloaded{longtable}{}{%
4   \PackageError{ltcaption}{longtable package not loaded, aborting}{RTFM}}
```

3.2 User interface

`\LTcapttype` `\LTcapttype` is preset to `table`.

```
5 \providecommand*\LTcapttype{table}
6 \providecommand*\ext@table{lot}
```

`\ext@lstlisting` Since the `listings` package do not define `\ext@lstlisting`, but we needed it when `\renewcommand\LTcapttype{lstlisting}` was done by the end user, we define it here.

```
7 \@ifundefined{caption@AtBeginDocument}\AtBeginDocument\caption@AtBeginDocument{%
8   \@ifpackageloaded{listings}{%
9     \providecommand*\ext@lstlisting{lol}}{}}
```

To save \TeX memory some stuff will not be defined if the caption package is loaded.

```
10 \@ifpackageloaded{caption}{}{%
```

`\LTcapskip`

```
11   \newskip\LTcapskip \LTcapskip=\abovecaptionskip
```

`\LTcapleft` Our skips and the flag belonging to them.

`\LTcapright` (Default: Use these skips (and not `\LTleft` & `\LTright`.)

`\ifLTcapmargins`

```
12   \newskip\LTcapleft \LTcapleft=\fill
13   \newskip\LTcapright \LTcapright=\fill
14   \newif\ifLTcapmargins \LTcapmarginstrue
```

`\CaptionLabelFont` These commands are provided by the NTG document classes. To make this package work with other document classes as well, we need to define `\CaptionLabelFont` & `\CaptionTextFont` here.

`\CaptionTextFont`

```
15   \providecommand*\CaptionLabelFont{}
16   \providecommand*\CaptionTextFont{}
```

`\CaptionLabelSeparator` Additionally, we define `\CaptionLabelSeparator` which is predefined as colon.

```
17   \providecommand*\CaptionLabelSeparator{:}
18 }
```

3.3 The longtable patch

`\LT@array` We insert our stuff into the definition of `\LT@array` here. 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`.

```
19 \@ifundefined{caption@AtBeginDocument}\AtBeginDocument\caption@AtBeginDocument{%
20   \let\ltcaption@ORI@LT@array\LT@array
21   \renewcommand*\LT@array{%
```

We modify `\refstepcounter` resp. `\H@refstepcounter` and `\hyper@makecurrent`, so `\LTcapttype` is used instead of `table`.

```

22 \let\caption@LT@refstepcounter\refstepcounter
23 \def\refstepcounter{%
24   \caption@LTtype\caption@LT@refstepcounter}%
25 \let\caption@LT@Hrefstepcounter\H@refstepcounter
26 \def\H@refstepcounter{%
27   \caption@LTtype\caption@LT@Hrefstepcounter}%
28 \let\caption@LT@makecurrent\hyper@makecurrent
29 \def\hyper@makecurrent{%
30   \caption@LTtype\caption@LT@makecurrent}%

```

We redefine `\lst@@caption` so `\thelstlisting` will printout its counter, too.

```

31 \def\lst@@caption{\relax}%
32 \ltcaption@ORI@LT@array}}
33 \newcommand*\caption@LTtype[2]{%
34   \edef\caption@LT@tempa{#2}%
35   \ifx\caption@LT@tempa\caption@LT@table
36     \caption@LT@type#1%
37   \else
38     #1{#2}%
39   \fi}%
40 \newcommand*\caption@LT@type[1]{%
41   \expandafter#1\expandafter{\LTcapttype}}
42 \newcommand*\caption@LT@table{table}%

```

`\LT@c@ption` The original implementation:

```

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

```

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

```

43 \long\def\LT@c@ption#1[#2]#3{%
44   \LT@makecaption#1{\csname fnum@\LTcapttype\endcsname}{#3}%
45   \def\@tempa{#2}%
46   \ifx\@tempa\@empty\else
47     {\let\\space
48      \addcontentsline{\csname ext@\LTcapttype\endcsname}{\LTcapttype}%
49      {\protect\numberline{\csname the\LTcapttype\endcsname}{#2}}}%
50   \fi
51   \ignorespaces}

```

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

Original code:

```

\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 code:¹

```

52 \renewcommand\LT@makecaption[3]{%
53   \LT@@makecaption{%
54     \sbox\@tempboxa{%
55       #1{\CaptionLabelFont#2\CaptionLabelSeparator} }\CaptionTextFont#3}%
56   \ifdim\wd\@tempboxa>\hsize
57     #1{\CaptionLabelFont#2\CaptionLabelSeparator} }\CaptionTextFont#3%
58   \else
59     \hbox to\hsize{\hfil\box\@tempboxa\hfil}%
60   \fi
61   \endgraf\vskip\LTcapskip}}

62 \newcommand\LT@@makecaption[1]{%
63   \caption@LT@make{\hb@xt@{\hsize{%
64     \ifLTcapwidth
65       \hspace\LTcapleft
66       \FBifLTcapwidth{\advance\LTcapwidth-\LTcapleft}%
67     \else
68       \hspace\LTleft
69       \FBifLTcapwidth{\advance\LTcapwidth-\LTleft}%
70     \fi
71     \FBifLTcapwidth{\advance\LTcapwidth-
72       \ifLTcapwidth\LTcapright\else\LTright\fi}%
73     \parbox[t]{\LTcapwidth}{#1}%
74     \ifLTcapwidth
75       \hskip\LTcapright
76     \else
77       \hskip\LTright
78     \fi}}}

```

This one will be usually defined by the `fr-longtable` package, which is part of the `floatrow` package^[4]:

```

79 \AtBeginDocument{\providecommand*\FBifLTcapwidth[1]{} }

```

`\caption@LT@make` Typesets the caption as `\multicolumn...`

```

80 \newcommand\caption@LT@make[1]{%
81   \noalign{\caption@LT@config}%

```

Note: If used with the `array` package `\caption@LTfmt` needs to be expanded, therefore we need some `\expandafter` here.

```

82   \expandafter\LT@mcol\expandafter\LT@cols\expandafter{\caption@LTfmt}{%

```

¹Adapted to the `floatrow` package by Olga Lapko

```

83     \hb@xt@\z@{%
84         \hspace\caption@LTleft
85         \parbox[t]{\linewidth{#1}}%
86         \hspace\caption@LTright}}}%

\caption@LT@config \caption@LT@config analyses \LTleft & \LTright and set \caption@LTleft
& \caption@LTright accordingly to the ‘opposite’ values, e.g., \LTleft=1cm
will result to \caption@LTleft=-1cm and \LTleft=0pt plus 1fill will re-
sult to \caption@LTleft=0pt minus 1fill. Furthermore \caption@LTfmt
is set to the according multicolumn format; this is far away from being bulletproof (e.g.,
a stretch or shrink will always be treated as ‘fill’) but will hopefully cover all ‘real’ cases.
87 \newcommand*\caption@LT@config{%
88     \caption@LT@parse\LTleft\caption@LTleft\caption@ifLTleft
89     \caption@LT@parse\LTright\caption@LTright\caption@ifLTright
90     \xdef\caption@LTfmt{%
91         @{}\caption@ifLTleft{\caption@ifLTright{c}{r}}{l}@{}}

\caption@LT@parse Parsing of the skip, we collect a \@fixpart, a @pluspart, and a \@minuspart
and make our definitions based on that.
92 \newcommand*\caption@LT@parse[3]{%
93     \let\@pluspart\@undefined
94     \let\@minuspart\@undefined
95     \expandafter\caption@LT@parse@\expandafter\@fixpart\the#1 x %
96     \xdef#2{-\@fixpart
97         \ifx\@minuspart\@undefined\else
98             \space\@plus\space\@minuspart
99         \fi
100        \ifx\@pluspart\@undefined\else
101            \space\@minus\space\@pluspart
102        \fi}%
103     \let#3\@firstoftwo
104     \ifx\@pluspart\@undefined
105         \ifx\@minuspart\@undefined
106             \let#3\@secondoftwo
107         \fi
108     \fi}

109 \def\caption@LT@parse@#1#2 {%

Note: \def\@tempa{#2}\ifx\@tempa\@plus... would not work here because of
different catcodes.
110     \edef\@tempa{\@car#2\@nil}%
111     \if p\@tempa
112         \def\next{\caption@LT@parse@\@pluspart}%
113     \else\if m\@tempa
114         \def\next{\caption@LT@parse@\@minuspart}%
115     \else\if x\@tempa
116         \let\next\relax
117     \else
118         \def#1{#2}%
119         \def\next{\caption@LT@parse@ @}%
120     \fi\fi\fi
121     \next}

```

3.4 The `longtable*` environment

`longtable*` A `longtable` environment without reference counter and hyperlink anchors.

```
122 \newenvironment{longtable*}{%
```

We simply suppress anything which has to do with reference counters here.

```
123 \let\caption@LT@type\@gobble
```

Unfortunately this is not so easy with `\Hy@raisedlink` (which actually sets the hyperlink anchor), so we assume that we have to suppress the first usage.

```
124 \let\caption@LT@raisedlink\Hy@raisedlink
```

```
125 \def\Hy@raisedlink{%
```

```
126 \let\Hy@raisedlink\caption@LT@raisedlink
```

```
127 \@gobble}%
```

Finally we are redefining `\caption` so the non-starred variant issues an error.

```
128 \let\lrcaption@ORI@LT@c@ption\LT@c@ption
```

```
129 \def\LT@c@ption##1[##2]##3{%
```

```
130 \ifx##1\@firstofone
```

```
131 \PackageError{lrcaption}%
```

```
132 {Not allowed in longtable* environment}%
```

```
133 {If you do not understand this error, please take a closer
```

```
134 look\MessageBreak at the documentation of the 'lrcaption'
```

```
135 package.\MessageBreak \@ehc}%
```

```
136 \else
```

```
137 \lrcaption@ORI@LT@c@ption{##1}[##2][##3}%
```

```
138 \fi}%
```

```
139 \longtable}%
```

```
140 {\endlongtable}
```

3.5 Adaption for KOMA-Script

```
141 \@ifundefined{@komalongtablefalse}{}{%
```

```
142 \if@komalongtable
```

```
143 \renewcommand{\LT@makecaption}[3]{%
```

```
144 \noalign{%
```

```
145 \if@captionabove
```

```
146 \vskip\belowcaptionskip
```

```
147 \else
```

```
148 \vskip\abovecaptionskip
```

```
149 \fi
```

```
150 }%
```

```
151 \caption@LT@make{%
```

```
152 \@@makecaption{#1}{#2}{#3}%
```

```
153 \endgraf
```

```
154 \if@captionabove
```

```
155 \vskip\abovecaptionskip
```

```
156 \else
```

```
157 \vskip\belowcaptionskip
```

```
158 \fi
```

```
159 }%
```

```
160 }%
```

```
161 \let\LT@@makecaption\@undefined
```

```
162 \fi}
```

References

- [1] David Carlisle: *The longtable package*, 2004/02/01
- [2] Victor Eijkhout: *An introduction to the Dutch L^AT_EX document classes*,
3 September 1989
- [3] Markus Kohm & Jens-Uwe-Morawski: *KOMA-Script – a versatile L^AT_EX 2_ε bundle*,
2007-01-09
- [4] Olga Lapko: *The floatrow package documentation*,
2007/08/24