

# The ltcaption package\*

Axel Sommerfeldt  
`caption@sommerfee.de`

2007/10/03

## Abstract

This package fixes caption problems with other-than-centered aligned longtables.  
(solves [L<sup>A</sup>T<sub>E</sub>X PR tools/3387](#))

## Contents

<b>1</b>	<b>The user interface</b>	<b>2</b>
1.1	Further justification . . . . .	2
1.2	Bonus features . . . . .	2
<b>2</b>	<b>Spot the difference</b>	<b>3</b>
<b>3</b>	<b>The Implementation</b>	<b>6</b>
3.1	Identification . . . . .	6
3.2	User interface . . . . .	6
3.3	The longtable patch . . . . .	6
3.4	The longtable* environment . . . . .	9
3.5	Adaption for KOMA-Script . . . . .	10

---

\*This package has version number v1.1, last revised 2007/09/01.

# 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

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

`\LTcapright`  
`\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 commands do not work when 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

A    B

`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 `lcaption` 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 `lcaption` 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 `lrcaption` 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 `lrcaption` 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 `ltablex` 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}[2007/09/01 v1.1 longtable captions (AR)]
```

### 3.2 User interface

`\LTcaptiontype` `\LTcaptiontype` is preset to `table`.

```
3 \providecommand*\LTcaptiontype{table}
4 \providecommand*\ext@table{lot}
```

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

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

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

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

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

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

```
\ifLTcapmargins
10 \newskip\LTcapleft \LTcapleft=\fill
11 \newskip\LTcapright \LTcapright=\fill
12 \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.

```
13 \providecommand*\CaptionLabelFont{}
14 \providecommand*\CaptionTextFont{}

15 }
```

### 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`.

```
16 \@ifundefined{caption@AtBeginDocument}\AtBeginDocument\caption@AtBeginDocument{%
17   \let\ltcaption@ORI@LT@array\LT@array
18   \renewcommand*\LT@array{%
```

We modify the next `\refstepcounter` resp. `\H@refstepcounter` & `\hyper@makecurrent`, so `\LTcaptiontype` is used instead of `{table}`.

```
19   \let\ltcaption@ORI@refstepcounter\refstepcounter
20   \let\ltcaption@ORI@H@refstepcounter\H@refstepcounter
21   \def\refstepcounter##1{%
```

```

22      \let\refstepcounter\lrcaption@ORI@refstepcounter
23      \let\H@refstepcounter\lrcaption@ORI@H@refstepcounter
24      \refstepcounter\LTcaption}%
25      \def\H@refstepcounter##1{%
26        \let\refstepcounter\lrcaption@ORI@refstepcounter
27        \let\H@refstepcounter\lrcaption@ORI@H@refstepcounter
28        \H@refstepcounter\LTcaption}%
29      \let\lrcaption@ORI@hyper@makecurrent\hyper@makecurrent
30      \def\hyper@makecurrent##1{%
31        \let\hyper@makecurrent\lrcaption@ORI@hyper@makecurrent
32        \hyper@makecurrent\LTcaption}%

```

We redefine \lst@@caption so \thelstlisting will print out its counter, too.

```

33      \def\lst@@caption{\relax}%
34      \lrcaption@ORI@LT@array}%
35 }

```

\LT@caption The original implementation:

```

\def\LT@caption#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 simply uses \LTcaption instead of {table}:

```

36 \def\LT@caption#1[#2]#3{%
37   \LT@makecaption#1{\csname fnun@\LTcaption\endcsname}{#3}%
38   \def\@tempa{#2}%
39   \ifx\@tempa\@empty\else
40     {\let\\\space
41      \addcontentsline{\csname ext@\LTcaption\endcsname}{\LTcaption}%
42      {\protect\numberline{\csname the\LTcaption\endcsname}{#2}}}%
43   \fi}

```

\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:<sup>1</sup>

```

44 \renewcommand\LT@makecaption[3]{%
45   \LT@makecaption{%
46     \sbox\@tempboxa{#1{\CaptionLabelFont#2:} }\CaptionTextFont#3}%
47     \ifdim\wd\@tempboxa>\hsize
48       #1{\CaptionLabelFont{#2:} }\CaptionTextFont#3%
49     \else
50       \hbox to\hsize{\hfil\box\@tempboxa\hfil}%
51     \fi
52   \endgraf\vskip\baselineskip}}

53 \newcommand\LT@makecaption[1]{%
54   \caption@LT@make{\hb@xt@\hsize{%
55     \ifLTcapmargin
56       \hspace\LTcapleft
57       \FBifLTcapwidth{\advance\LTcapwidth-\LTcapleft}%
58     \else
59       \hspace\LTleft
60       \FBifLTcapwidth{\advance\LTcapwidth-\LTleft}%
61     \fi
62     \FBifLTcapwidth{\advance\LTcapwidth-%
63       \ifLTcapmargin\LTcapright\else\LTright\fi}%
64     \parbox[t]{\LTcapwidth{#1}%
65       \ifLTcapmargin
66         \hskip\LTcapright
67       \else
68         \hskip\LTright
69       \fi}}}

```

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

```

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

```

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

```

71 \newcommand\caption@LT@make[1]{%
72   \noalign{\caption@LT@config}%

```

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

```

73   \expandafter\LT@mcol\expandafter\LT@cols\expandafter{\caption@LTfmt}{%
74     \hb@xt@\z@{%
75       \hspace\caption@LTleft
76       \parbox[t]{\linewidth{#1}%
77       \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 result 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.

```

78 \newcommand*\caption@LT@config{%

```

---

<sup>1</sup>Adapted to the `floatrow` package by Olga Lapko



```

79 \caption@LT@parse\LTleft\caption@LTleft\caption@ifLTleft
80 \caption@LT@parse\LTRight\caption@LTRight\caption@ifLTRight
81 \xdef\caption@LTfmt{%
82   @{} \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.

```

83 \newcommand*\caption@LT@parse[3]{%
84   \let\@pluspart\@undefined
85   \let\@minuspart\@undefined
86   \expandafter\caption@LT@parse@\expandafter\@fixpart\the#1 x %
87   \xdef#2{-\@fixpart
88     \ifx\@minuspart\@undefined\else
89       \space\@plus\space\@minuspart
90     \fi
91     \ifx\@pluspart\@undefined\else
92       \space\@minus\space\@pluspart
93     \fi}%
94   \let#3\@firstoftwo
95   \ifx\@pluspart\@undefined
96     \ifx\@minuspart\@undefined
97       \let#3\@secondoftwo
98     \fi
99   \fi}
100 \def\caption@LT@parse@#1#2 {%

```

Note: \def\@tempa{#2}\ifx\@tempa\@plus... would not work here because of different catcodes.

```

101 \edef\@tempa{\@car#2\@nil}%
102 \if p\@tempa
103   \def\next{\caption@LT@parse@\@pluspart}%
104 \else\if m\@tempa
105   \def\next{\caption@LT@parse@\@minuspart}%
106 \else\if x\@tempa
107   \let\next\relax
108 \else
109   \def#1{#2}%
110   \def\next{\caption@LT@parse@ @}%
111 \fi\fi\fi
112 \next}

```

### 3.4 The longtable\* environment

longtable\* We simply suppress anything which has to do with reference counters or hyper links here.

```

113 \newenvironment{longtable*}%
114   {\let\refstepcounter\@gobble
115    \let\H@refstepcounter\@gobble
116    \let\hyper@makecurrent\@gobble
117    \let\Hy@raisedlink\@gobble
118    \let\ltcaption@ORI@LT@caption\LT@caption
119    \def\LT@caption##1[##2]##3{%

```

```

120 \ifx##1\@firstofone
121 \PackageError{ltcaption}%
122 {Not allowed in longtable* environment}%
123 {If you do not understand this error, please take a closer
124 look\MessageBreak at the documentation of the 'ltcaption'
125 package.\MessageBreak \@ehc}%
126 \else
127 \ltcaption@ORI@LT@c@ption{##1}[{##2}]{##3}%
128 \fi}%
129 \longtable}%
130 {\endlongtable}

```

### 3.5 Adaption for KOMA-Script

```

131 \@ifundefined{@komalongtablefalse}{}{%
132 \if@komalongtable
133 \renewcommand{\LT@makecaption}[3]{%
134 \noalign{%
135 \if@captionabove
136 \vskip\belowcaptionskip
137 \else
138 \vskip\abovecaptionskip
139 \fi
140 }%
141 \caption@LT@make{%
142 \@@makecaption{#1}{#2}{#3}%
143 \endgraf
144 \if@captionabove
145 \vskip\abovecaptionskip
146 \else
147 \vskip\belowcaptionskip
148 \fi
149 }%
150 }%
151 \let\LT@@makecaption\@undefined
152 \fi}

```

## References

- [1] David Carlisle: *The longtable package*, 2004/02/01
- [2] Victor Eijkhout: *An introduction to the Dutch L<sup>A</sup>T<sub>E</sub>X document classes*, 3 September 1989
- [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] Olga Lapko: *The floatrow package documentation*, 2007/08/24