The longtable package[*]

David Carlisle[†]

2004/02/01

Abstract

This package defines the longtable environment, a multi-page version of tabular.

List of Tables

1 An optional table caption (used in the list of tables) 1
2 A floating table 4
3 A difficult `\multicolumn` combination: pass 1 6
4 A difficult `\multicolumn` combination: pass 2 6
5 A difficult `\multicolumn` combination: pass 3 6
6 A difficult `\multicolumn` combination: pass 4 6
7 A summary of longtable commands 9

1 Introduction

`longtable` The longtable package defines a new environment, longtable, which has most of the features of the tabular environment, but produces tables which may be broken by TeX's standard page-breaking algorithm. It also shares some features with the table environment. In particular it uses the same counter, `table`, and has a similar `\caption` command. Also, the standard `\listoftables` command lists tables produced by either the table or longtable environments.

The following example uses most of the features of the longtable environment. An edited listing of the input for this example appears in Section 8.

Note: Various parts of the following table will **not** line up correctly until this document has been run through LaTeX several times. This is a characteristic feature of this package, as described below.

Table 1: A long table

*	This part appears at the top of the table		*
*	FIRST	SECOND	*
*	longtable columns are specified	in the	*
*	same way as in the tabular	environment.	*
*	This goes at the	bottom.	*

[*] This file has version number v4.11, last revised 2004/02/01.

[†] The new algorithm for aligning 'chunks' of a table used in version 4 of this package was devised coded and documented by David Kastrup, `dak@neuroinformatik.ruhr-uni-bochum.de`.

Table 1: (continued)

*	This part appears at the top of every other page		*
*	**First**	**Second**	*
*	@{*}r||p{1in}@{*}	in this case.	*
*	Each row ends with a	\\ command.	*
*	The \\ command has an	optional	*
*	argument, just as in	the	*
*	tabular	environment.	*
*	See the effect of \\[10pt]	?	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Also \hline may be used,	as in tabular.	*
*	That was a \hline	.	*
*	That was \hline\hline	.	*
‖	This is a \multicolumn{2}{||c||}		‖
*	If a page break occurs at a \hline then	a line is drawn	*
*	at the bottom of one page and at the	top of the next.	*
*	The [t] [b] [c] argument of tabular	can not be used.	*
*	The optional argument may be one of	[l] [r] [c]	*
*	to specify whether the table should be	adjusted	*
*	to the left, right	or centrally.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	This goes at the	bottom.	*

Table 1: (continued)

*	This part appears at the top of every other page		*
*	**First**	**Second**	*
*	Some lines may take up a lot of space, like this:	This last column is a “p” column so this “row” of the table can take up several lines. Note however that TeX will never break a page within such a row. Page breaks only occur between rows of the table or at `\hline` commands.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	Lots[1] of lines	like this.	*
*	Lots of lines	like this[2]	*
*	Lots of lines	like this.	*
*	Lots of lines	like this.	*
*	These lines will	appear	*
*	in place of the	usual foot	*
*	at the end	of the table	*

2 Chunk Size

`LTchunksize`

In order to TeX multi-page tables, it is necessary to break up the table into smaller chunks, so that TeX does not have to keep everything in memory at one time. By default longtable uses 20 rows per chunk, but this can be set by the user, with e.g., `\setcounter{LTchunksize}{10}`.[3] These chunks do not affect page breaking, thus if you are using a TeX with a lot of memory, you can set `LTchunksize` to be several pages of the table. TeX will run faster with a large `LTchunksize`. However, if necessary, longtable can work with `LTchunksize` set to 1, in which

[1] This is a footnote.

[2] longtable takes special precautions, so that footnotes may also be used in ‘p’ columns.

[3] You can also use the plain TeX syntax `\LTchunksize=10`.

A	tabular	environment
within	a floating	table

Table 2: A floating table

case the memory taken up is negligible. Note that if you use the commands for setting the table head or foot (see below), the `LTchunksize` must be at least as large as the number of rows in each of the head or foot sections.

This document specifies `\setcounter{LTchunksize}{10}`. If you look at the previous table, after the *first* run of LaTeX you will see that various parts of the table do not line up. LaTeX will also have printed a warning that the column widths had changed. longtable writes information onto the `.aux` file, so that it can line up the different chunks. Prior to version 4 of this package, this information was not used unless a `\setlongtables` command was issued, however, now the information is always used, using a new algorithm[4] and so `\setlongtables` is no longer needed. It is defined (but does nothing) for the benefit of old documents that use it.

3 Captions and Headings

`\endhead` `\endfirsthead` `\endfoot` `\endlastfoot`

At the start of the table one may specify lines which are to appear at the top of every page (under the headline, but before the other lines of the table). The lines are entered as normal, but the last `\\` command is replaced by a `\endhead` command. If the first page should have a different heading, then this should be entered in the same way, and terminated with the `\endfirsthead` command. The `LTchunksize` should be at least as large as the number of rows in the heading. There are also `\endfoot` and `\endlastfoot` commands which are used in the same way (at the *start* of the table) to specify rows (or an `\hline`) to appear at the bottom of each page. In certain situations, you may want to place lines which logically belong in the table body at the end of the firsthead, or the beginning of the lastfoot. This helps to control which lines appear on the first and last page of the table.

`\caption`

The `\caption{...}` command is essentially equivalent to
`\multicolumn{n}{c}{\parbox{\LTcapwidth}{...}}`
where `n` is the number of columns of the table. You may set the width of the caption with a command such as `\setlength{\LTcapwidth}{2in}` in the preamble of your document. The default is 4in. `\caption` also writes the information to produce an entry in the list of tables. As with the `\caption` command in the figure and table environments, an optional argument specifies the text to appear in the list of tables if this is different from the text to appear in the caption. Thus the caption for table 1 was specified as `\caption[An optional table caption (used in the list of tables)]{A long table\label{long}}`.

You may wish the caption on later pages to be different to that on the first page. In this case put the `\caption` command in the first heading, and put a subsidiary caption in a `\caption[]` command in the main heading. If the optional argument to `\caption` is empty, no entry is made in the list of tables. Alternatively, if you do not want the table number to be printed each time, use the `\caption*`

[4] Due to David Kastrup.

command.

The captions are set based on the code for the article class. If you have redefined the standard `\@makecaption` command to produce a different format for the captions, you may need to make similar changes to the longtable version, `\LT@makecaption`. See the code section for more details.

A more convenient method of customising captions is given by the caption(2) package, which provides commands for customising captions, and arranges that the captions in standard environments, and many environments provided by packages (including longtable) are modified in a compatible manner.

You may use the `\label` command so that you can cross reference longtables with `\ref`. Note however, that the `\label` command should not be used in a heading that may appear more than once. Place it either in the firsthead, or in the body of the table. It should not be the *first* command in any entry.

4 Multicolumn entries

The `\multicolumn` command may be used in longtable in exactly the same way as for tabular. So you may want to skip this section, which is rather technical, however coping with `\multicolumn` is one of the main problems for an environment such as longtable. The main effect that a user will see is that certain combinations of `\multicolumn` entries will result in a document needing more runs of LaTeX before the various 'chunks' of a table align.

The examples in this section are set with `LTchunksize` set to the minimum value of one, to demonstrate the effects when `\multicolumn` entries occur in different chunks.

Consider Table 3. In the second chunk, longtable sees the wide multicolumn entry. At this point it thinks that the first two columns are very narrow. All the width of the multicolumn entry is assumed to be in the third column. (This is a 'feature' of TeX's primitive `\halign` command.) longtable then passes the information that there is a wide third column to the later chunks, with the result that the first pass over the table is too wide.

If the 'saved row' from this first pass was re-inserted into the table on the next pass, the table would line up in two passes, but would be much two wide.

`\kill` The solution to this problem used in Versions 1 and 2, was to use a `\kill` line. If a line is `\kill`ed, by using `\kill` rather than `\\` at the end of the line, it is used in calculating column widths, but removed from the final table. Thus entering `\kill`ed copies of the last two rows before the wide multicolumn entry would mean that `\halign` 'saw' the wide entries in the first two columns, and so would not widen the third column by so much to make room for the multicolumn entry.

In Version 3, a new solution was introduced. If the saved row in the `.aux` file was not being used, longtable used a special 'draft' form of `\multicolumn`, this modified the definition, so the spanning entry was never considered to be wider than the columns it spanned. So after the first pass, the `.aux` file stored the widest normal entry for each column, no column was widened due to `\spanned` columns. By default longtable ignored the `.aux` file, and so each run of LaTeX was considered a first pass. Once the `\setlongtables` declaration was given, the saved row in the `.aux` file, and the proper definition of `\multicolumn` were used. If any `\multicolumn` entry caused one of the columns to be widened, this

Table 3: A difficult `\multicolumn` combination: pass 1

1	2	3
wide multicolumn spanning 1–3		
multicolumn 1–2		3
wide 1	2	3

Table 4: A difficult `\multicolumn` combination: pass 2

1	2	3
wide multicolumn spanning 1–3		
multicolumn 1–2		3
wide 1	2	3

Table 5: A difficult `\multicolumn` combination: pass 3

1	2	3
wide multicolumn spanning 1–3		
multicolumn 1–2		3
wide 1	2	3

Table 6: A difficult `\multicolumn` combination: pass 4

1	2	3
wide multicolumn spanning 1–3		
multicolumn 1–2		3
wide 1	2	3

information could not be passed back to earlier chunks, and so the table would not correctly line up until the third pass. This algorithm always converged in three passes as described above, but in examples such as the ones in Tables 3–6, the final widths were not optimal as the width of column 2, which is determined by a `\multicolumn` entry was not known when the final width for column 3 was fixed, due to the fact that *both* `\multicolumn` commands were switched from 'draft' mode to 'normal' mode at the same time.

Version 4 alleviates the problem considerably. The first pass of the table will indeed have the third column much too wide. However, on the next pass longtable will notice the error and reduce the column width accordingly. If this has to propagate to chunks before the `\multicolumn` one, an additional pass will, of course, be needed. It is possible to construct tables where this rippling up of the correct widths taks several passes to 'converge' and produce a table with all chunks aligned. However in order to need many passes one needs to construct a table with many overlapping `\multicolumn` entries, all being wider than the natural widths of the columns they span, and all occuring in different chunks. In the typical case the algorithm will converge after three or four passes, and, the benefits of not needing to edit the document before the final run to add `\setlongtables`, and the better choice of final column widths in the case of multiple `\multicolumn` entries will hopefully more than pay for the extra passes that may possibly be needed.

So Table 3 converges after 4 passes, as seen in Table 6.

You can still speed the convergence by introducing judicious `\kill` lines, if you happen to have constellations like the above.

If you object even to LaTeX-ing a file twice, you should make the first line of every longtable a `\kill` line that contains the widest entry to be used in each column. All chunks will then line up on the first pass.

5 Adjustment

The optional argument of longtable controls the horizontal alignment of the table. The possible options are `[c]`, `[r]` and `[l]`, for centring, right and left adjustment, respectively. Normally centring is the default, but this document specifies

\LTleft
\LTright

```
\setlength\LTleft\parindent
\setlength\LTright\fill
```

in the preamble, which means that the tables are set flush left, but indented by the usual paragraph indentation. Any lengths can be specified for these two parameters, but at least one of them should be a rubber length so that it fills up the width of the page, unless rubber lengths are added between the columns using the `\extracolsep` command. For instance

```
\begin{tabular*}{\textwidth}{@{\extracolsep{...}}...}
```

produces a full width table, to get a similar effect with longtable specify

```
\setlength\LTleft{0pt}
\setlength\LTright{0pt}
\begin{longtable}{@{\extracolsep{...}}...}
```

6 Changes

This section highlights the major changes since version 2. A more detailed change log may be produced at the end of the code listing if the `ltxdoc.cfg` file specifies

```
\AtBeginDocument{\RecordChanges}
\AtEndDocument{\PrintChanges}
```

Changes made between versions 2 and 3.

- The mechanism for adding the head and foot of the table has been completely rewritten. With this new mechanism, longtable does not need to issue a `\clearpage` at the start of the table, and so the table may start half way down a page. Also the `\endlastfoot` command which could not safely be implemented under the old scheme, has been added.

- longtable now issues an error if started in the scope of `\twocolumn`, or the multicols environment.

- The separate documentation file `longtable.tex` has been merged with the package file, `longtable.dtx` using Mittelbach's doc package.

- Support for footnotes has been added. Note however that `\footnote` will not work in the 'head' or 'foot' sections of the table. In order to put a footnote in those sections (e.g., inside a caption), use `\footnotemark` at that point, and `\footnotetext` anywhere in the table *body* that will fall on the same page.

- The treatment of `\multicolumn` has changed, making `\kill` lines unnecessary, at the price of sometimes requiring a third pass through LaTeX.

- The `\newpage` command now works inside a longtable.

Changes made between versions 3 and 4.

- A new algorithm is used for aligning chunks. As well as the widest width in each column, longtable remembers which chunk produced this maximum. This allows it to check that the maximum is still achieved in later runs. As longtable can now deal with columns shrinking as the file is edited, the `\setlongtables` system is no longer needed and is disabled.

- An extra benefit of the new algorithm's ability to deal with 'shrinking' columns is that it can give better (narrower) column widths in the case of overlapping `\multicolumn` entries in different chunks than the previous algorithm produced.

- The 'draft' multicolumn system has been removed, along with related commands such as `\LTmulticolumn`.

- The disadvantage of the new algorithm is that it can take more passes. The theoretical maximum is approximately twice the length of a 'chain' of columns with overlapping `\multicolumn` entries, although in practice it usually converges as fast as the old version. (Which always converged in three passes once `\setlongtables` was activated.)

- `\\*` and `\nopagebreak` commands may be used to control page breaking.

7 Summary

Table 7: A summary of longtable commands

Parameters		
`\LTleft`	Glue to the left of the table.	(`\fill`)
`\LTright`	Glue to the right of the table.	(`\fill`)
`\LTpre`	Glue before the the table.	(`\bigskipamount`)
`\LTpost`	Glue after the the table.	(`\bigskipamount`)
`\LTcapwidth`	The width of a parbox containing the caption.	(4in)
`LTchunksize`	The number of rows per chunk.	(20)

Optional arguments to `\begin{longtable}`	
none	Position as specified by `\LTleft` and `\LTright`.
`[c]`	Centre the table.
`[l]`	Place the table flush left.
`[r]`	Place the table flush right.

Commands to end table rows	
`\\`	Specifies the end of a row
`\\[`⟨*dim*⟩`]`	Ends row, then adds vertical space (as in the tabular environment).
`\\*`	The same as `\\` but disallows a page break after the row.
`\tabularnewline`	Alternative to `\\` for use in the scope of `\raggedright` and similar commands that redefine `\\`.
`\kill`	Row is ‘killed’, but is used in calculating widths.
`\endhead`	Specifies rows to appear at the top of every page.
`\endfirsthead`	Specifies rows to appear at the top the first page.
`\endfoot`	Specifies rows to appear at the bottom of every page.
`\endlastfoot`	Specifies rows to appear at the bottom of the last page.

longtable caption commands	
`\caption{`⟨*caption*⟩`}`	Caption ‘Table ?: ⟨*caption*⟩’, and a ‘⟨*caption*⟩’ entry in the list of tables.
`\caption[`⟨*lot*⟩`]{`⟨*caption*⟩`}`	Caption ‘Table ?: ⟨*caption*⟩’, and a ‘⟨*lot*⟩’ entry in the list of tables.
`\caption[]{`⟨*caption*⟩`}`	Caption ‘Table ?: ⟨*caption*⟩’, but no entry in the list of tables.
`\caption*{`⟨*caption*⟩`}`	Caption ‘⟨*caption*⟩’, but no entry in the list of tables.

Commands available at the start of a row	
`\pagebreak`	Force a page break.
`\pagebreak[`⟨*val*⟩`]`	A ‘hint’ between 0 and 4 of the desirability of a break.
`\nopagebreak`	Prohibit a page break.
`\nopagebreak[`⟨*val*⟩`]`	A ‘hint’ between 0 and 4 of the undesirability of a break.
`\newpage`	Force a page break.

Footnote commands available inside longtable	
`\footnote`	Footnotes, but may not be used in the table head & foot.
`\footnotemark`	Footnotemark, may be used in the table head & foot.
`\footnotetext`	Footnote text, use in the table body.

Setlongtables	
`\setlongtables`	Obsolete command. Does nothing now.

8 Verbatim highlights from Table 1

```
\begin{longtable}{@{*}r||p{1in}@{*}}
KILLED & LINE!!!! \kill
\caption[An optional table caption ...]{A long table\label{long}}\\
\hline\hline
\multicolumn{2}{@{*}c@{*}}%
     {This part appears at the top of the table}\\
\textsc{First}&\textsc{Second}\\
\hline\hline
\endfirsthead
\caption[]{(continued)}\\
\hline\hline
\multicolumn{2}{@{*}c@{*}}%
      {This part appears at the top of every other page}\\
\textbf{First}&\textbf{Second}\\
\hline\hline
\endhead
\hline
This goes at the&bottom.\\
\hline
\endfoot
\hline
These lines will&appear\\
in place of the & usual foot\\
at the end& of the table\\
\hline
\endlastfoot
\env{longtable}  columns  are specified& in the \\
same way as  in the \env{tabular}& environment.\\
...
\multicolumn{2}{||c||}{This is a ...}\\
...
Some lines may take...&
    \raggedleft This last column is a ``p'' column...
    \tabularnewline
...
Lots of lines& like this.\\
...
\hline
Lots\footnote{...} of lines& like this.\\
Lots   of   lines& like this\footnote{...}\\
\hline
Lots of lines& like this.\\
...
\end{longtable}
```