Making Tables with Macros

Ray F. Cowan
22 February 1985

Tables have traditionally been difficult to make using TEX – especially ruled tables. The file `TABLES.TEX` contains macros designed to prepare both ruled and unruled tables with considerably less effort. Note that `TABLES.TEX` can be used with any macro set; it does not depend on prior loading of `PHYZZX` or any other macro set, for example. One of the main advantages of this macro set is that you no longer need to design a preamble for the table; the macros will scan your table entries and construct a suitable preamble for you. To access the macros, say `‘\input TABLES’` in your TEX file, somewhere before you first use them. The macros available are listed in Table 1 and Table 2.

Macro name	*Description*
`\begintable`	Indicates the start of a new table
`\endtable`	Ends the current table. Must be used in the place of the last `\cr`.
`\cr`	Ends the current row, and starts the next one. The completed row will be separated from the next with a thin horizontal rule.
`\crthick`	Similar to `\cr`, but the rows will be separated with a thick horizontal rule.
`\crnorule` or `\nr`	Similar to `\cr`, but the rows will not be separated by any rule.
`|` (vertical bar) or `\vb`	Separates one column from the next, and puts a vertical rule between them.
`&` or `\novb`	Same as `|`, but does not put in the vertical rule between the columns.
`|`	Same as `|`, but puts in a thick vertical rule.

Table 1: Description of simple table macros

`\thicksize`=*dimen*	This dimension specifies the thickness of the thick rules in the table. The default size is 1.5 points.
`\thinsize`=*dimen*	This dimension specifies the thickness of thin rules in the table. The default size is 0.8 points.
`\tablewidth`=*dimen*	Specifies how wide to make the next table. If not specified, the table is made to its natural width. This value is reset following the construction of each table.
`\multispan`{*n*}	Makes the next entry span the next n columns, where n is an integer, $n > 0$. See other notes on `\multispan` below.
`\omit`	This TeX primitive causes the normal template for its entry to be omitted, allowing the user to do something else with this entry.
`\para`{*paragraph text*}	Turns an entry into a neat little paragraph like this one. The width of the paragraph is determined by the dimension `\parasize`. The default is 4 inches.
`\parasize`=*dimen*	Sets the width of paragraphs produced with the `\para` macro.
`\ctr{#}`	Used in the standard template, this macro centers its argument in the column. The user can redefine it for special effects. The default definition is `\def\ctr#1{\hfil\ #1\ \hfil}`
`\vctr{#}`	Centers an entry vertically. The vertical center of the entry is placed on the baseline of the row containing it. The intended use is to center an entry between two rows.
`\centeredtables`	Turns table centering on. Each table will be centered left-to-right on the page. This is the default.
`\noncenteredtables`	Turns table centering off. Each table is returned as an `\hbox`, and it is up to the user to place it as desired.
`\tableinfotrue`	Turns on the diagnostic message telling you how many rows and columns were found in the table. This is the default.
`\tableinfofalse`	Turns off the diagnostic messages concerning rows and columns.

Table 2: Description of extended table macros

The general idea is that you start your table with the command `\begintable`, type your entries in one row at a time, then finish with the command `\endtable`. To specify a row, enter the individual entries into your TEX file, separating each column entry with a `|`, an `&`, or a `\|`. A `|` will separate the adjoining columns with a thin vertical rule, an `&` will leave out the vertical rule, and a `\|` will separate the columns with a thick vertical rule. To end one row and start another, use either a `\cr`, a `\crnorule`, or a `\crthick`. A `\cr` separates the rows with a thin horizontal rule; a `\crnorule` leaves out the horizontal rule, while `\crthick` inserts a thick horizontal rule. Then end the last row with an `\endtable`.

Each row of the table must contain the same number of columns, otherwise unpredictable things will happen. Again the *last row* must *not* end with `\cr...`, but *must* end with an `\endtable`. If you put a `\cr` and an `\endtable` both on the last row, you won't like what happens.

Each entry will be centered in it's column (unless you use `\omit`, a TEX primitive, or `\multispan`. See notes below). The table will be centered in a `\hbox` of width `\hsize`, unless you have turned table centering off (see the commands `\centeredtables` and `\noncenteredtables`).

Each time a new table is encountered, a message similar to '`[Nrows=xx, Ncols=yy]`' is printed on your terminal, where `xx` is the number of rows and `yy` the number of columns discovered in your table. If you think they are incorrect, you may have left out some `|`'s or `&`'s or `\cr`'s. This diagnostic feature can be disabled by saying `\tableinfofalse` (and restarted by saying `\tableinfotrue`).

An example

A simple 3-row, 2-column table with a header spanning two columns could be specified as (see notes 3 and 4 below on the use of `\multispan`):

```
\begintable
\multispan{2}\tstrut\hfil The Top Line\hfil\crthick
Entry 1 | Entry 2 \cr
Entry 3 | Entry 4 \endtable
```

These commands produce Table 3.

The Top Line	
Entry 1	Entry 2
Entry 3	Entry 4

Table 3: A sample table

An example of non-centered tables

Two or more tables can be placed side-by-side by using the `\noncenteredtables` command. Consider the two tables here (Table 4):

Item ABC	Item DEF
Item GHI	Item JKL

Data 111	Data 222
Data 333	Data 444
Data 555	Data 666

Table 4: Two non-centered tables aligned side-by-side

These were produced by saying

```
\noncenteredtables
\line{
\begintable
Item ABC | Item DEF \cr
Item GHI | Item JKL \endtable
\hfil
\begintable
Data 111 | Data 222 \cr
Data 333 | Data 444 \cr
Data 555 | Data 666 \endtable
}
```

Notice that tables of unequal height are aligned at the bottom.

Usage notes:

1. Vertical spacing is done with a strut, called `\tstrut`, which is initially defined as 3.1ex high and 1.2ex deep. If you don't like the way it looks, you can redefine `\tstrut` to your own liking:

   ```
   \def\tstrut{\vrule height hh depth dd width 0pt}
   ```

   where you specify your desired height `hh` and depth `dd`.

2. You can control the thickness of the thin and thick rules through the use of `\thicksize` and `\thinsize`. You can turn off the rules entirely by saying

   ```
   \thinsize=0pt
   \thicksize=0pt
   ```

   for example

3. If you use a `\multispan` or an `\omit` in the first column of a row, you will lose the effect of the `\tstrut` within that row and must specify it yourself. See the example above.
4. Use of `\multispan` and `\omit` will cause the default centering of the entry to be lost; if you want it centered, put an `\hfil` on each side of the entry, as in the example above.
5. To override the default centering action, include an `\hfill` on the left or right as desired; the `\hfill` will override the default `\hfil`.

Local modifications for use with LaTeX:

This macro package was originally written for use with plain TeX. Bob Taylor has made changes for it to be used as a style option in LaTeX. The plain TeX command `\line` has been replaced with `\tableline`, with `\tableline` being defined as

```
\def\tableline{\hbox to \hsize}
```

The `\documentstyle` command in your LaTeX file should look something like:

```
\documentstyle[tables]{article}
```