

The `tabulary` package*

David Carlisle

2024/06/04

1 User Documentation

`\begin{tabulary}{\langle length \rangle}{\langle pream \rangle} ... \end{tabulary}`

The rather daft name may change in a later release but it is a pun on `tabularx`, which itself was a pun on `tabular*`...

These environments work pretty much like the standard `tabular` environment (or more correctly, the enhanced version from the `array` package) except that there are more possibilities for the column types.

LCRJ These new ‘uppercase’ column types are only activated in the `tabulary` environment. In order to make the total table width equal to $\langle length \rangle$ the LCRJ columns are converted to `p` columns (with `\raggedright`, `\centering`, or `\raggedleft` or normal justification respectively applied). The width of these converted columns is proportional to the natural width of the longest entry in each column.

To stop very narrow columns being too ‘squeezed’ by this process any columns that are narrower than `\tymax` are set to their natural width. This length may be set with `\setlength` and is arbitrarily initialised to 10 pt. (If you know that a column will be narrow, it may be preferable to use, say, `c` rather than `C` so that the `tabulary` mechanism is never invoked on that column.)

Similarly one very large entry can force its column to be too wide. So to prevent this, all columns with natural length greater than `\tymax` are set to the same width (with the proportion being taken as if the natural length was *equal* to `\tymax`). This is initially set to twice the text width..

Narrow `p` columns are sometimes quite hard to set, and so you may redefine the command `\tyformat` to be any declarations to make just after the `\centering` or `\ragged...` declaration. By default it redefines `\everypar` to insert a zero space at the start of every paragraph, so the first word may be hyphenated. (See `DogBook`).

As the environment makes a standard L^AT_EX box, it will be indented by the paragraph indent at the start of a paragraph, and so will not fit on a line if given argument `\textwidth` unless it is preceded by `\noindent` or is in a `center` environment or some other environment with zero paragraph indent.

*This file has version number v0.11, last revised 2024/06/04.

2 Features

You can use `\multicolumn` but if the multicolumn text turns out to be longer than the final calculated widths of the columns that it spans, then the final table will be too wide.

`\verb` doesn't work. (except in restricted version as in `tabularx`)

The whole table is evaluated twice, so take care with some \TeX constructions that may have side effects like writing to files.

3 Options

The following package option is defined:

debugshow Causes a lot of stuff to appear on the terminal. I find this invaluable, you may find it less so.

4 Examples

With C columns

1	the rain in spain	(an @ expr.)	the rain in spain falls mainly on the
	falls mainly on the		plain the rain in spain falls mainly on
	plain		the plain
a	b	(an @ expr.)	c
a	b b	(an @ expr.)	c c
a			

With J columns

1	the rain in spain	(an @ expr.)	the rain in spain falls mainly on the
	falls mainly on the		plain the rain in spain falls mainly on
	plain		the plain
a	b	(an @ expr.)	c
a	b b	(an @ expr.)	c c
a			

With L, R and C columns, and a `\multicolumn`

1	the rain in spain	the rain in spain falls mainly on	and now for
	falls mainly on	the plain the rain in spain falls	something
	the plain	mainly on the plain	completely
			different
x		some multicolumn text across columns 2-4	
a	b	c	d
a	b b	c c	d d
a			

The following examples attempt to show the effect of the `\tymax` and `\tymax` parameters. One should also perhaps note that `\tymax` refers to the total column width (including any inter-column space, rules etc) but `\tymax` just refers to the width of the column entry (like the argument to the standard `p` column).

`\tymax=0pt`
`\tymax=\maxdimen`

Note how the first column is ‘squeezed’. In fact it is in such a narrow column that even ‘a’ produces an overfull box warning!

a	b	c c c c c	d d d d d d d d d d d d d d d d d d d
b	b	c c c c c	d d d d d d d d d d d d d d d d d d d
b	b	c c c c c	d d d d d d d d d d d d d d d d d d d
	b	c c c c c	d d d d d d d d d d d d d d d d d d d
		c c c c c	d d d d d d d d d d d d d d d d d d d
		c c	d d d d d d d d d d d d d d d d d d d

`\tymax=20pt`
`\tymax=\maxdimen`

Here increase `\tymax` so that columns b and a are not so narrow. ‘a’ is set to its natural width, and ‘b’ is set to `\tymax`.

a	b b b	c c c c	d d d d d d d d d d d d d d d d d d d
	b	c c c c	d d d d d d d d d d d d d d d d d d d
		c c c c	d d d d d d d d d d d d d d d d d d d
		c c c c	d d d d d d d d d d d d d d d d d d d
		c c c c	d d d d d d d d d d d d d d d d d d d
		c c c c	d d d d d d d d d d d d d d d d d d d
		c c c c	d d d d d d d d d d d d d d d d d d d

`\tymax=20pt`
`\tymax=200pt`

In the previous example, the large d column dominated the table, being a lot wider than the c column. By reducing `\tymax` can limit the width of column d producing more even column widths, but now producing an entry for d that is longer than that for c.

a	b b b	c c c c c c c c c c	d d d d d d d d d d
	b	c c c c c c c c c c	d d d d d d d d d d
		c c c c c c c c c c	d d d d d d d d d d
			d d d d d d d d d d
			d d d d d d d d d d
			d d d d d d d d d d
			d d d d d d d d d d
			d d d d d d d d d d
			d d d d d d d d d d
			d d d d d d d d d d
			d d d d d d d d d d

5 The Code

```

1 (*package)
   Rollback.
2 \DeclareRelease{v0.10}{1995/10/08}{tabulary-v010.sty}
3 \DeclareCurrentRelease{}{2024-06-01}
   This version needs a current array package
4 \RequirePackage{array}[2024/05/23]
5 \catcode'\Z=14
6 \DeclareOption{debugshow}{\catcode'\Z=9\relax}
7 \ProcessOptions

\arraybackslash Borrowed from tabularx.
   8 \def\arraybackslash{\let\=\@arraycr}

\@finalstrut Bug fixed version from December 1995 LATEX release. Old bug going back to
LATEX2.09...
   9 \def\@finalstrut#1{%
10   \unskip\ifhmode\nobreak\fi\vrule\@width\z@\@height\z@\@depth\dp#1}

\TY@count Counter so that we know what column we are hacking around in.
11 \newcount\TY@count

\tabulary Top level macro for standard form.
12 \def\tabulary{%
13   \let\TY@final\tabular
14   \let\endTY@final\endtabular
15   \TY@tabular}

\TY@tabular Looks a lot like tabularx at this stage. Grab everything into a token register.
16 \def\TY@tabular#1{%
17   \edef\TY@{\@currenvir}%
18   {\ifnum0='}\fi

At this point need to save locally things that tabulary will globally mess up. These
are restored at the end of the environment.
19   \@ovxx\TY@linewidth
20   \@ovyy\TY@tablewidth
21   \count@\z@
22   \@tempwattrue
23   \@whilesw\if@tempwa\fi{%
24     \advance\count@\@ne
25     \expandafter\ifx\csname TY@F\the\count@\endcsname\relax
26       \@tempwafalse
27     \else
28       \expandafter\let\csname TY@SF\the\count@\endcsname
29         \csname TY@F\the\count@\endcsname
30       \global\expandafter\let\csname TY@F\the\count@\endcsname\relax
31       \expandafter\let\csname TY@S\the\count@\endcsname
32         \csname TY@\the\count@\endcsname
33   \fi}%

```

```

34 \global\TY@count\@ne
35 \TY@width\xdef{Opt}%
36 \global\TY@tablewidth\z@
37 \global\TY@linewidth#1\relax
38 Z\message{^^J^^JTable^^J%
39 Z      Target Width: \the\TY@linewidth^^J%
40 Z      \string\tabcolsep: \the\tabcolsep\space
41 Z      \string\arrayrulewidth: \the\arrayrulewidth\space
42 Z      \string\doublerulesep: \the\doublerulesep^^J%
43 Z      \string\tymin: \the\tymin\space
44 Z      \string\tymax: \the\tymax^^J}%

```

Placing this here means that nested tabulars will get this definition but that's probably OK, the extra code for LCR etc shouldn't do any harm

```

45 \let\@classz\TY@classz
46 \let\verb\TX@verb
47 \toks@{\}\TY@get@body}

```

\TY@mkpream Saved version.

```
48 \let\TY@mkpream\@mkpream
```

\TY@mkpream TY version.

```

49 \ExplSyntaxOn
50 \def\TY@mkpream{%
51   \def\@addamp{%
52     \if@firstamp \@firstampfalse \else
53       \global\advance\TY@count\@ne
54     \edef\@preamble{\@preamble & \noexpand\tbl_update_cell_data:}\fi
55     \TY@width\xdef{Opt}}%
56   \def\@acol{%
57     \TY@subwidth\col@sep
58     \@addtopreamble{\hskip\col@sep}}%
59   \let\@arrayrule\TY@arrayrule
60   \let\@classvi\TY@classvi
61   \def\@classv{\save@decl
62     \expandafter\NC@ecs\@nextchar\extracolsep{}\extracolsep\@@_tbl
63     \sbox\z@\d@llarbegin\@nextchar\d@llarend}%
64     \TY@subwidth{\wd\z@}%
65     \@addtopreamble{\d@llarbegin\the@toks\the\count@relax\d@llarend}%
66     \prepnext@tok}%
67   \global\let\@mkpream\TY@mkpream
68   \TY@mkpream}
69 \ExplSyntaxOff

```

\TY@arrayrule Pull this out so the colortbl support below can redefine

```

70 \def\TY@arrayrule{%
71   \TY@subwidth\arrayrulewidth
72   \@addtopreamble \vline}

```

`\TY@classvi` Pull this out so the colortbl support below can redefine

```
73 \def\TY@classvi{\ifcase \@lastchclass
74 \@acol \or
75 \TY@subwidth\doublerulesep
76 \@addtopreamble{\hskip \doublerulesep}\or
77 \@acol \or
78 \classvii
79 \fi}
```

`\TY@tab` First run a tabular with all the column types fudged so that the widths of any rules or @-expressions are noted.

```
80 \def\TY@tab{%
81 \setbox\z@\hbox\bgroup

Support displaymath by making it non-display in the first pass. (Other display
environments defined in terms of $$ would need to be added here by packages that
define them.)

82 \let\[$\let\]$%
83 \let\equation$\let\endequation$%

84 \col@sep\tabcolsep
85 \let\d@llarbegin\begin\group\let\d@llarend\end\group

86 \let\@mkpream\TY@mkpream

87 \def\multicolumn##1##2##3{\multispan##1\relax}%
88 \CT@start\TY@tabarray}
```

`\TY@tabarray`

```
89 \def\TY@tabarray{\@ifnextchar[{\TY@array}{\@array[t]}}
90 \def\TY@array[#1]{\@array[t]}
```

`\TY@width` Just a shorthand to access a column width macro.

```
91 \def\TY@width#1{%
92 \expandafter#1\csname TY@the\TY@count\endcsname}
```

`\TY@subwidth` Subtract a width from the current column width and also The total line table width and the target line width.

```
93 \def\TY@subwidth#1{%
94 \TY@width\dimen@
95 \advance\dimen@-#1\relax
96 \TY@width\xdef{the\dimen@}%
97 \global\advance\TY@linewidth-#1\relax}
```

`\endtabulary` First run one modified tabular, making sure to add a blank row (cf longtable) to the end in case the user supplied last row is hidden by an hline or something.

```
98 \def\endtabulary{%
99 \SuspendTagging {tabulary}%
100 \gdef\@halignto{}}

Save values of counters, to reset after the trial

101 \def\@elt##1{\global\value{##1}\the\value{##1}\relax}%
102 \edef\TY@ckpt{\cl@ckpt}%
```

```

103 \expandafter\TY@tab\the\toks@
104 \crrc\omit
105 {\xdef\TY@save@row{}}%
106   \loop
107   \advance\TY@count\m@ne
108   \ifnum\TY@count>\z@
109   \xdef\TY@save@row{\TY@save@row&\omit}%
110   \repeat\TY@save@row
111 \endarray\global\setbox1=\lastbox\setbox0=\vbox{\unvbox1
112 \unskip\global\setbox1=\lastbox}\egroup
113 \ResumeTagging {tabulary}%

```

Check that \tymin is not too large.

```

114 \dimen@\TY@linewidth
115 \divide\dimen@\TY@count
116 \ifdim\dimen@<\tymin
117   \TY@warn{tymin too large (\the\tymin), resetting to \the\dimen@}%
118   \tymin\dimen@
119 \fi

```

Now take the last row apart, cf longtable or appendix D.

```

120 \setbox\tw@=\hbox{\unhbox\@ne
121   \loop
122   \@tempdima=\lastskip
123   \ifdim\@tempdima>\z@
124   Z   \message{ecs=\the\@tempdima^^J}%
125   \global\advance\TY@linewidth-\@tempdima
126   \fi
127   \unskip
128   \setbox\tw@=\lastbox
129   \ifhbox\tw@
130   Z   \message{Col \the\TY@count: Initial=\the\wd\tw@\space}%
131   \ifdim\wd\tw@>\tymax
132     \wd\tw@\tymax
133   Z   \message{> max\space}%
134   Z   \else
135   Z   \message{ \@spaces\space}%
136   \fi
137   \TY@width\dimen@
138   Z \message{\the\dimen@\space}%
139   \advance\dimen@\wd\tw@
140   Z \message{Final=\the\dimen@\space}%
141   \TY@width\xdef{\the\dimen@}%
142   \ifdim\dimen@<\tymin
143   Z   \message{< tymin}%
144   \global\advance\TY@linewidth-\dimen@
145   \expandafter\xdef\csname TY@F\the\TY@count\endcsname
146   \the\dimen@}%
147   \else
148   \expandafter\ifx\csname TY@F\the\TY@count\endcsname\z@
149   Z   \message{***}%
150   \global\advance\TY@linewidth-\dimen@
151   \expandafter\xdef\csname TY@F\the\TY@count\endcsname
152   \the\dimen@}%
153   \else

```



```

154 Z      \message{> tymin}%
155      \global\advance\TY@tablewidth\dimen@
156      \global\expandafter\let\csname TY@F\the\TY@count\endcsname
157      \maxdimen
158      \fi\fi
159      \advance\TY@count\m@ne
160      \repeat}%

```

A bit cheap just doing this four times, but prevents any possibilities of looping...

```

161      \TY@checkmin
162      \TY@checkmin
163      \TY@checkmin
164      \TY@checkmin

```

Reset the counter.

```
165      \TY@count\z@
```

Reset the LCRJ column definition to set paragraphs to the calculated widths.

```
166      \let\TY@box\TY@box@v
```

Restore counter values.

```
167      \TY@ckpt
```

Run a second tabular, and for the star form, unbox it.

```
168      {\expandafter\TY@final\the\toks@endTY@final}%

```

Finish off by restoring global commands.

```

169      \count@\z@
170      \@tempwattrue
171      \@whiles\if@tempwa\fi{%
172      \advance\count@\@ne
173      \expandafter\ifx\csname TY@SF\the\count@\endcsname\relax
174      \@tempwafalse
175      \else
176      \global\expandafter\let\csname TY@F\the\count@\expandafter\endcsname
177      \csname TY@SF\the\count@\endcsname
178      \global\expandafter\let\csname TY@\the\count@\expandafter\endcsname
179      \csname TY@S\the\count@\endcsname
180      \fi}%
181      \TY@linewidth\@ovxx
182      \TY@tablewidth\@ovyy
183      \ifnum0='{ \fi}

```

`\TY@checkmin` Check that no column is squeezed below `\tymin`. If it is, fix the width of that column to `\tymin` and try again re-computing the ratio. (The new ratio will be smaller, and may squeeze yet more rows, so need to iterate this, currently just do it four times.)

```

184 \def\TY@checkmin{%
185   \let\TY@checkmin\relax
186   \ifdim\TY@tablewidth>\z@
187     \Gscale@div\TY@ratio\TY@linewidth\TY@tablewidth
188     % \changes{v0.9}{2008/12/01}
189     %   {\cs{TY@linewidth}}
190   \ifdim\TY@tablewidth <\TY@linewidth
191     \def\TY@ratio{1}%

```

```

192 \fi
193 \else
194 \TY@warn{No suitable columns!}%
195 \def\TY@ratio{1}%
196 \fi
197 \count@\z@
198 Z \message{^^JLine Width: \the\TY@linewidth,
199 Z           Natural Width: \the\TY@tablewidth,
200 Z           Ratio: \TY@ratio^^J}%
201 \@tempdima\z@
202 \loop
203 \ifnum\count@<\TY@count
204 \advance\count@\@ne
205 \ifdim\csname TY@F\the\count@\endcsname>\tymin
206 \dimen@\csname TY@\the\count@\endcsname
207 \dimen@\TY@ratio\dimen@
208 \ifdim\dimen@<\tymin
209 Z \message{Column \the\count@\space ->}%
210 \global\expandafter\let\csname TY@F\the\count@\endcsname\tymin
211 \global\advance\TY@linewidth-\tymin
212 \global\advance\TY@tablewidth-\csname TY@\the\count@\endcsname
213 \let\TY@checkmin\TY@@checkmin
214 \else
215 \expandafter\xdef\csname TY@F\the\count@\endcsname{\the\dimen@}%
216 \advance\@tempdima\csname TY@F\the\count@\endcsname
217 \fi
218 \fi
219 Z \dimen@\csname TY@F\the\count@\endcsname\message{\the\dimen@, }%
220 \repeat
221 Z \message{^^JTotal:\the\@tempdima^^J}%
222 }

\TY@@checkmin Saved value
223 \let\TY@@checkmin\TY@checkmin

\TY@linewidth Stores the target width.
224 \newdimen\TY@linewidth

\tyformat What to do with columns
225 \def\tyformat{\everypar{\nobreak\hskip\z@skip}}

tymin Columns narrower than this are not fudged.
226 \newdimen\tymin
227 \tymin=10pt

tymin Columns wider than this are all treated alike and set to the same width, to stop
one particularly long entry hijacking the entire table.
228 \newdimen\tymax
229 \tymax=2\textwidth

```

`\@testpatch` Also add LCRJ although these don't do anything useful except in tabulary.

```
230 \def\@testpach{\@chclass
231 \ifnum \@lastchclass=6 \@ne \@chnum \@ne \else
232 \ifnum \@lastchclass=7 5 \else
233 \ifnum \@lastchclass=8 \tw@ \else
234 \ifnum \@lastchclass=9 \thr@@
235 \else \z@
236 \ifnum \@lastchclass = 10 \else
237 \edef\@nextchar{\expandafter\string\@nextchar}%
238 \@chnum
239 \if \@nextchar c\z@ \else
240 \if \@nextchar l\@ne \else
241 \if \@nextchar r\tw@ \else
242 % \if \@nextchar s6 \else
243 \if \@nextchar C7 \else
244 \if \@nextchar L8 \else
245 \if \@nextchar R9 \else
246 \if \@nextchar J10 \else
247 \z@ \@chclass
248 \if\@nextchar |\@ne \else
249 \if \@nextchar !6 \else
250 \if \@nextchar @7 \else
251 \if \@nextchar <8 \else
252 \if \@nextchar >9 \else
253 10
254 \@chnum
255 \if \@nextchar m\thr@@\else
256 \if \@nextchar p4 \else
257 \if \@nextchar b5 \else
258 \z@ \@chclass \z@ \@preamerr \z@ \fi \fi \fi \fi \fi \fi \fi \fi
259 % \fi
260 \fi \fi \fi \fi \fi \fi \fi \fi \fi \fi \fi}
```

`\TY@classz` Here hacked around without the respect Frank's code deserves...

```
261 \def\TY@classz{%
262 \@classx
263 \@tempcnta\count@
264 \ifx\TY@box\TY@box@v
265 \global\advance\TY@count\@ne
266 \fi
267 \let\centering c%
268 \let\raggedright\noindent
269 \let\raggedleft\indent
270 \let\arraybackslash\relax
271 \prepnext@tok
272 \ifnum\@chnum<6
273 \global\expandafter\let\csname TY@F\the\TY@count\endcsname\z@
274 \fi
275 \ifnum\@chnum=6
276 \global\expandafter\let\csname TY@F\the\TY@count\endcsname\z@
277 \fi
278 \@addtopreamble{%
279 \ifcase\@chnum
280 \hfil\hskip1sp%
```

```

281 \d@llarbegin\insert@column\d@llarend\do@row@strut\hfil \or
282 \hskip1sp%
283 \d@llarbegin \insert@column \d@llarend\do@row@strut\hfil \or
284 \hfil\hskip1sp%
285 \d@llarbegin \insert@column \d@llarend\do@row@strut \or
286 \setbox\ar@mcellbox\vbox
287 \@startpbox{\@nextchar}\insert@pcolumn \@endpbox
288 \ar@align@mcell
289 \do@row@strut\or
290 \vtop \@startpbox{\@nextchar}\insert@pcolumn \@endpbox \or
291 \vbox \@startpbox{\@nextchar}\insert@pcolumn \@endpbox \or
292 \d@llarbegin \insert@column \d@llarend \do@row@strut \or% dubious "s" case
293 \TY@box\centering\or
294 \TY@box\raggedright\or
295 \TY@box\raggedleft\or
296 \TY@box\relax
297 \fi}\prepnext@tok}

```

`\TY@box` The argument is `\centering` etc depending on whether LCRJ is used. However in this version the entries are set in horizontal mode with definitions mimicing the standard lcr columns. Later `\TY@box` will be redefined to `\TY@box@v` which really sets the entries in vertical mode.

```

298 \def\TY@box#1{%
299 \ifx\centering#1%
300 \hfil\hskip1sp%
301 \d@llarbegin\insert@column\d@llarend\do@row@strut \hfil \else
302 \ifx\raggedright#1%
303 \hskip1sp%
304 \d@llarbegin \insert@column \d@llarend\do@row@strut \hfil \else
305 \ifx\raggedleft#1%
306 \hfil\hskip1sp%
307 \kern\z@ \d@llarbegin \insert@column \d@llarend\do@row@strut \else
308 \ifx\relax#1%
309 \d@llarbegin \insert@column \d@llarend\do@row@strut
310 \fi \fi \fi \fi}

```

`\TY@box@v` The version to use in a final run, set the CLRJ columns in a parbox of the appropriate width.

```

311 \def\TY@box@v#1{%
312 \vtop \@startpbox{\csname TY@F\the\TY@count\endcsname}%
313 #1\arraybackslash\tyformat
314 \insert@pcolumn\@endpbox}

```

`\TY@tablewidth` The natural width of the table on the first run.

```

315 \newdimen\TY@tablewidth

```

`\Gscale@div` Stolen from graphics package.

```

316 \def\Gscale@div#1#2#3{%
317 \setlength\dimen@{#3}%
318 \ifdim\dimen@=\z@
319 \PackageError{graphics}{Division by 0}\@eha
320 \dimen@#2%
321 \fi

```

```

322 \edef\@tempd{\the\dimen@}%
323 \setlength\dimen@{#2}%
324 \count@65536\relax
325 \ifdim\dimen@<\z@
326   \dimen@-\dimen@
327   \count@-\count@
328 \fi
329 \loop
330   \ifdim\dimen@<8192\p@
331     \dimen@\tw@\dimen@
332     \divide\count@\tw@
333 \repeat
334 \dimen@ii=\@tempd\relax
335 \divide\dimen@ii\count@
336 \divide\dimen@\dimen@ii
337 \edef#1{\strip@pt\dimen@}

```

`\TY@get@body` Place all tokens as far as the first `\end` into a token register. Then call `\TY@find@end` to see if we are at `\end{tabulary}`.

```

338 \long\def\TY@get@body#1\end
339   {\toks@\expandafter{\the\toks@#1}\TY@find@end}

```

`\TY@find@end` If we are at `\end{tabulary}`, call `\end{tabulary}`, otherwise add `\end{...}` to the register, and call `\TY@get@body` again.

```

340 \def\TY@find@end#1{%
341   \def\@tempa{#1}%
342   \ifx\@tempa\TY@\def\@tempa{\end{#1}}\expandafter\@tempa
343   \else\toks@\expandafter
344     {\the\toks@\end{#1}}\expandafter\TY@get@body\fi}

```

`\TY@warn` Warning messages.

```

345 \def\TY@warn{%
346   \PackageWarning{tabulary}}

```

```

347 \catcode'\Z=11

```

colortbl support.

```

348 \AtBeginDocument{
349   \@ifpackageloaded{colortbl}{%
350     \expandafter\def\expandafter\@mkpream\expandafter#\expandafter1%
351     \expandafter{%
352       \expandafter\let\expandafter\CT@setup\expandafter\relax
353       \expandafter\let\expandafter\CT@color\expandafter\relax
354       \expandafter\let\expandafter\CT@do@color\expandafter\relax
355       \expandafter\let\expandafter\color\expandafter\relax
356       \expandafter\let\expandafter\CT@column@color\expandafter\relax
357       \expandafter\let\expandafter\CT@row@color\expandafter\relax
358       \expandafter\let\expandafter\CT@cell@color\expandafter\relax
359       \@mkpream{#1}}
360   \let\TY@\@mkpream\@mkpream
361   \def\TY@classz{%
362     \@classx
363     \@tempcnta\count@
364     \ifx\TY@box\TY@box@v

```



```

418 \ifx\CT@drsc@\relax
419 \addtopreamble{\hskip\doublerulesep}%
420 \else
421 \addtopreamble{\CT@drsc@\vrule\width\doublerulesep}}%
422 \fi\or
423 \@acol \or
424 \@classvii
425 \fi}%
426 }{%
427 \let\CT@start\relax
428 }
end of at begin document
429 }

```

`\TX@warn` `\verb` support, uses same csnames as in TX so they share code if both loaded (this version names tabulary in the warning though). See tabularx for documentation.

```

430 {\uccode'\*='\ %
431 \uppercase{\gdef\TX@verb{%
432 \leavevmode\null\TX@vwarn
433 {\ifnum0='}\fi\ttfamily\let\\\ignorespaces
434 \@ifstar{\let~*\TX@vb}{\TX@vb}}}}
435 \def\TX@vb#1{\def\@tempa##1#1{\toks@{##1}\edef\@tempa{\the\toks@}%
436 \expandafter\TX@v\meaning\@tempa\ \ \ \ifnum0='{ \fi}}\@tempa!}
437 \def\TX@v#1!{\afterassignment\TX@vfirst\let\@tempa= }
438 \begingroup
439 \catcode'\*=\catcode'\#
440 \catcode'\#=12
441 \gdef\TX@vfirst{%
442 \if\@tempa#%
443 \def\@tempb{\TX@v@#}%
444 \else
445 \let\@tempb\TX@v@
446 \if\@tempa\space~\else\@tempa\fi
447 \fi
448 \@tempb}
449 \gdef\TX@v@*1 *2{%
450 \TX@v@hash*1#\relax\if*2\\\else~\expandafter\TX@v@\fi*2}
451 \gdef\TX@v@hash*1##*2{*1\ifx*2\relax\else#\expandafter\TX@v@hash\fi*2}
452 \endgroup
453 \def\TX@vwarn{%
454 \@warning{\noexpand\verb may be unreliable inside tabularx/y}%
455 \global\let\TX@vwarn\@empty}

```

`\tblcrern` Patch to ensure `\tbl_crern:n` ends with `\crern`. (Will be fixed in later L^AT_EX releases.

```

456 (package)
457 \ExplSyntaxOn
458 \cs_set:Npn \@tempa #1 {
459 \int_compare:nNnT \g__tbl_col_int > 0
460 {
461 \tbl_count_missing_cells:n {#1}
462 \cr

```

```

463     }
464   }
465 \ifx\@tempa\tbl_crcr:n
466 \cs_set:Npn \tbl_crcr:n #1 {
467   \int_compare:nNnT \g__tbl_col_int > 0
468   {
469     \tbl_count_missing_cells:n {#1}
470   }
471   \crcr
472 }
473 \fi
474 \let\@tempa\@undefined
475 \ExplSyntaxOff
476 </package>

```

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in *roman* refer to the code lines where the entry is used.

	Symbols		
\#	439, 440	\@firstampfalse	52 312, 388, 391, 392
*	430, 439	\@halignto	100
\@@	62	\@height	10, 409 436, 437, 442,
\@acol	56, 74, 77, 416, 423	\@ifnextchar	89 446, 458, 465, 474
\@addamp	51	\@ifpackageloaded	349
\@addtopreamble	58, 65, 72, 76, 278, 379, 414, 419, 421	\@ifstar	434
\@array	89, 90	\@lastchclass	73, 231, 232, 233, 234, 236, 415
\@arraycr	8	\@mkpream	48, 67, 86, 350, 359, 360
\@arrayrule	59	\@ne	24, 34, 53, 120, 172, 204, 231, 240, 248, 265, 365
\@chclass	230, 247, 258	\@nextchar	62, 63, 237, 239, 240, 241, 242, 243, 244, 245, 246, 248, 249, 250, 251, 252, 255, 256, 257, 287, 290, 291, 388, 391, 392
\@chnum	231, 238, 254, 272, 275, 279, 373, 376, 381	\@nil	372
\@classv	61	\@ovxx	19, 181
\@classvi	60	\@ovyy	20, 182
\@classvii	78, 424	\@preamble	54
\@classx	262, 362	\@preamerr	258
\@classz	45	\@spaces	135
\@currenvir	17	\@startpbox	287, 290, 291,
\@depth	10		
\@eha	319		
\@elt	101		
\@empty	455		
\@endpbox	287, 290, 291, 314, 388, 391, 392		
\@finalstrut	<u>9</u>		
		\@tempa	341, 342, 435, 436, 437, 442,
		\@tempb	443, 445, 448
		\@tempcnta	263, 363, 372
		\@tempd	322, 334
		\@tempdima	122, 123, 124, 125, 201, 216, 221, 407, 408, 409
		\@tempswafalse	26, 174
		\@tempswatruer	22, 170
		\@testpach	230
		\@testpatch	<u>230</u>
		\@undefined	474
		\@warning	454
		\@whilesw	23, 171
		\@width	10, 409, 421
		\[82
		\]	8, 433, 436, 450
		\l	430
		\]	82
		A	
		\advance	24, 53, 95, 97, 107, 125, 139, 144, 150, 155, 159, 172, 204, 211, 212, 216, 265, 365, 408

<code>\afterassignment</code> .. 437	<code>\CT@setup</code> 352, 401	<code>\egroup</code> 112, 399
<code>\ar@align@mcell</code> 288, 389	<code>\CT@start</code> 88, 427	<code>\else</code> 27, 52, 134, 147,
<code>\ar@mcellbox</code> .. 286, 387		153, 175, 193,
<code>\arraybackslash</code> ...	D	214, 231, 232,
.. 8, 270, 313, 370	<code>\d@llarbegin</code> ... 63,	233, 235, 236,
<code>\arrayrulewidth</code> ...	65, 85, 281, 283,	239, 240, 241,
..... 41, 71, 413	285, 292, 301,	242, 243, 244,
<code>\AtBeginDocument</code> .. 348	304, 307, 309,	245, 246, 248,
	383, 385, 386, 393	249, 250, 251,
B	<code>\d@llarend</code> 63,	252, 255, 256,
<code>\begingroup</code> 85, 400, 438	65, 85, 281, 283,	257, 301, 304,
<code>\bgroup</code> 81, 380	285, 292, 301,	307, 343, 420,
	304, 307, 309,	444, 446, 450, 451
	383, 385, 386, 393	<code>\end</code> 338, 342, 344
C	<code>\DeclareCurrentRelease</code>	<code>\endarray</code> 111
<code>\catcode</code> 3	<code>\endcsname</code> 25,
5, 6, 347, 439, 440	<code>\DeclareOption</code> 6	28, 29, 30, 31,
<code>\centering</code> ... 267,	<code>\DeclareRelease</code> 2	32, 92, 145, 148,
293, 299, 367, 394	<code>\def</code> 8,	151, 156, 173,
<code>\changes</code> 188	9, 12, 16, 50, 51,	176, 177, 178,
<code>\cl@ckpt</code> 102	56, 61, 70, 73,	179, 205, 206,
<code>\col@sep</code> 57, 58, 84	80, 87, 89, 90,	210, 212, 215,
<code>\color</code> 355	91, 93, 98, 101,	216, 219, 273,
<code>\columncolor</code> 372	184, 191, 195,	276, 312, 374, 377
<code>\count@</code> .. 21, 24, 25,	225, 230, 261,	<code>\endequation</code> 83
28, 29, 30, 31,	298, 311, 316,	<code>\endgroup</code> . 85, 406, 452
32, 65, 169, 172,	338, 340, 341,	<code>\endtabular</code> 14
173, 176, 177,	342, 345, 350,	<code>\endtabulary</code> 98
178, 179, 197,	361, 412, 415,	<code>\endTY@final</code> ... 14, 168
203, 204, 205,	435, 437, 443, 453	<code>\equation</code> 83
206, 209, 210,	<code>\dimen@</code> 94, 95, 96, 114,	<code>\everypar</code> 225
212, 215, 216,	115, 116, 117,	<code>\expandafter</code>
219, 263, 324,	118, 137, 138,	. 25, 28, 30, 31,
327, 332, 335, 363	139, 140, 141,	62, 92, 103, 145,
<code>\cr</code> 462	142, 144, 146,	148, 151, 156,
<code>\crrc</code> 104, 471	150, 152, 155,	168, 173, 176,
<code>\cs</code> 189, 458, 466	206, 207, 208,	178, 210, 215,
<code>\csname</code> 25,	215, 219, 317,	237, 273, 276,
28, 29, 30, 31,	318, 320, 322,	339, 342, 343,
32, 92, 145, 148,	323, 325, 326,	344, 350, 351,
151, 156, 173,	330, 331, 336, 337	352, 353, 354,
176, 177, 178,	<code>\dimen@ii</code> . 334, 335, 336	355, 356, 357,
179, 205, 206,	<code>\divide</code> 115, 332, 335, 336	358, 372, 374,
210, 212, 215,	<code>\do@row@strut</code>	377, 436, 450, 451
216, 219, 273,	. 281, 283, 285,	<code>\ExplSyntaxOff</code> . 69, 475
276, 312, 374, 377	289, 292, 301,	<code>\ExplSyntaxOn</code> .. 49, 457
<code>\CT@arc@</code> 414	304, 307, 309, 390	<code>\extracolsep</code> 62
<code>\CT@cell@color</code> 358, 404	<code>\doublerulesep</code> . 42,	
<code>\CT@color</code> 353	75, 76, 417, 419, 421	F
<code>\CT@column@color</code> ..	<code>\dp</code> 10	<code>\fi</code> 10, 18, 23,
..... 356, 402		33, 54, 79, 119,
<code>\CT@do@color</code> .. 354, 405	E	126, 136, 158,
<code>\CT@drsc@</code> 418, 421	<code>\edef</code> ... 17, 54, 102,	171, 180, 183,
<code>\CT@extract</code> 372	237, 322, 337, 435	192, 196, 217,
<code>\CT@row@color</code> . 357, 403		

218, 258, 259,	236, 272, 275,	
260, 266, 274,	373, 376, 433, 436	
277, 297, 310,	\ifx ... 25, 148, 173,	
321, 328, 344,	264, 299, 302,	
366, 375, 378,	305, 308, 342,	
398, 422, 425,	364, 418, 451, 465	
433, 436, 446,	\ignorespaces ... 433	
447, 450, 451, 473	\indent ... 269, 369	
	\insert@column 281,	
	283, 285, 292,	
	301, 304, 307,	
	309, 383, 385, 386	
	\insert@pcolumn 287,	
	290, 291, 314,	
	388, 391, 392, 393	
	\int ... 459, 467	
	K	
	\kern . 307, 382, 384, 386	
	L	
\Gscale@div ... 187, <u>316</u>	\lastbox .. 111, 112, 128	
	\lastskip ... 122	
	\leavevmode ... 432	
	\let 8, 13, 14, 28, 30,	
	31, 45, 46, 48,	
	59, 60, 67, 82,	
	83, 85, 86, 156,	
	166, 176, 178,	
	185, 210, 213,	
	223, 267, 268,	
	269, 270, 273,	
	276, 352, 353,	
	354, 355, 356,	
	357, 358, 360,	
	367, 368, 369,	
	370, 374, 377,	
	427, 433, 434,	
	437, 445, 455, 474	
	\long ... 338	
	\loop . 106, 121, 202, 329	
	M	
	\m@ne ... 107, 159	
	\maxdimen ... 157	
	\meaning ... 436	
	\message ... 38,	
	124, 130, 133,	
	135, 138, 140,	
	143, 149, 154,	
	198, 209, 219, 221	
	\minrowclearance .. 408	
	\multicolumn ... 87	
	\multispan ... 87	
	N	
	\NC@ecs ... 62	
	\newcount ... 11	
	\newdimen ...	
	. 224, 226, 228, 315	
	\nobreak ... 10, 225	
	\noexpand ... 54, 454	
	\noindent ... 268, 368	
	\null ... 432	
	O	
	\omit ... 104, 109	
	\or . 74, 76, 77, 281,	
	283, 285, 289,	
	290, 291, 292,	
	293, 294, 295,	
	383, 385, 386,	
	390, 391, 392,	
	393, 394, 395,	
	396, 416, 422, 423	
	P	
	\p@ ... 330	
	\PackageError ... 319	
	\PackageWarning ... 346	
	\prepnext@tok .. 66,	
	271, 297, 371, 411	
	\ProcessOptions ... 7	
	R	
	\raggedleft ... 269,	
	295, 305, 369, 396	
	\raggedright .. 268,	
	294, 302, 368, 395	
	\relax ... 6, 25,	
	30, 37, 65, 87,	
	95, 97, 101, 173,	
	185, 270, 296,	
	308, 324, 334,	
	352, 353, 354,	
	355, 356, 357,	
	358, 370, 397,	
	418, 427, 450, 451	
	\repeat 110, 160, 220, 333	
	\RequirePackage ... 4	
	\ResumeTagging ... 113	
	S	
	\save@decl ... 61	
	\sbox ... 63	
	\setbox ... 81,	
	111, 112, 120,	
	128, 286, 380, 387	
	\setlength ... 317, 323	

<code>\space</code> . . .	40, 41, 43, 130, 133, 135, 138, 140, 209, 446	<code>\TX@vwarn</code> .	432, 453, 455	<code>\TY@warn</code> . . .	117, 194, <u>345</u>
<code>\stretch</code>	382, 383	<code>\TX@warn</code>	<u>430</u>	<code>\TY@width</code> . . .	35, 55, <u>91</u> , 94, 96, 137, 141
<code>\string</code>	40, 41, 42, 43, 44, 237	<code>\TY@</code>	17, 342	<code>\tyformat</code>	<u>225</u> , 313
<code>\strip@pt</code>	337	<code>\TY@@checkmin</code> .	<u>213</u> , <u>223</u>	<code>\tymax</code>	44, 131, 132, 228, 229
<code>\SuspendTagging</code> . . .	99	<code>\TY@@mkpream</code> <u>48</u> , 67, 68, 360	<code>\tymin</code> .	43, 116, 117, 118, 142, 205, 208, 210, 211, <u>226</u> , 226, 227, <u>228</u>
T					
<code>\tabcolsep</code>	40, 84	<code>\TY@array</code>	89, 90		
<code>\tabular</code>	13	<code>\TY@arrayrule</code> 59, <u>70</u> , 412			
<code>\tabulary</code>	<u>12</u>	<code>\TY@box</code> . . .	166, 264, 293, 294, 295, 296, <u>298</u> , 364, 394, 395, 396, 397		
<code>\tbl</code> 54, 461, 465, 466, 469		<code>\TY@box@v</code> 166, 264, <u>311</u> , 364	U	
<code>\tblcrn</code>	<u>456</u>	<code>\TY@checkmin</code> 161, 162,	163, 164, <u>184</u> , 223	<code>\uccode</code>	430
<code>\textwidth</code>	<u>229</u>	<code>\TY@ckpt</code>	102, 167	<code>\unhbox</code>	120, 410
<code>\the</code> . .	25, 28, 29, 30, 31, 32, 39, 40, 41, 42, 43, 44, 65, 92, 96, 101, 103, 117, 124, 130, 138, 140, 141, 145, 146, 148, 151, 152, 156, 168, 173, 176, 177, 178, 179, 198, 199, 205, 206, 209, 210, 212, 215, 216, 219, 221, 273, 276, 312, 322, 339, 344, 372, 374, 377, 435	<code>\TY@classvi</code> .	60, <u>73</u> , 415	<code>\unskip</code>	10, 112, 127
<code>\the@toks</code>	65	<code>\TY@classz</code> 45, <u>261</u> , 361		<code>\unvbox</code>	111
<code>\thr@@</code>	234, 255	<code>\TY@count</code> . . .	<u>11</u> , 34, 53, 92, 107, 108, 115, 130, 145, 148, 151, 156, 159, 165, 203, 265, 273, 276, 312, 365, 374, 377	<code>\uppercase</code>	431
<code>\toks</code>	372	<code>\TY@final</code>	13, 168	V	
<code>\toks@</code> .	47, 103, 168, 339, 343, 344, 435	<code>\TY@find@end</code> . .	339, <u>340</u>	<code>\value</code>	101
<code>\ttfamily</code>	433	<code>\TY@get@body</code> 47, <u>338</u> , 344		<code>\vbox</code>	111, 286, 291, 387, 392
<code>\tw@</code> 120, 128, 129, 130, 131, 132, 139, 233, 241, 331, 332		<code>\TY@linewidth</code> . .	19, 37, 39, 97, 114, 125, 144, 150, 181, 187, 190, 198, 211, <u>224</u> , 224	<code>\verb</code>	46, 454
<code>\TX@v</code>	436, 437	<code>\TY@linewidht</code>	19, 37, 39, 97, 114, 125, 144, 150, 181, 187, 190, 198, 211, <u>224</u> , 224	<code>\vline</code>	72, 414
<code>\TX@v@</code> 443, 445, 449, 450		<code>\TY@final</code>	13, 168	<code>\vrule</code>	10, 409, 421
<code>\TX@v@hash</code> . . .	450, 451	<code>\TY@find@end</code> . .	339, <u>340</u>	<code>\vtop</code>	290, 312, 391
<code>\TX@vb</code>	434, 435	<code>\TY@get@body</code> 47, <u>338</u> , 344		W	
<code>\TX@verb</code>	46, 431	<code>\TY@linewidth</code> . .	19, 37, 39, 97, 114, 125, 144, 150, 181, 187, 190, 198, 211, <u>224</u> , 224	<code>\wd</code> 64, 130, 131, 132, 139	
<code>\TX@vfirst</code> . . .	437, 441	<code>\TY@linewidht</code>	19, 37, 39, 97, 114, 125, 144, 150, 181, 187, 190, 198, 211, <u>224</u> , 224	X	
		<code>\TY@mkpream</code>	49, 86	<code>\xdef</code>	35, 55, 96, 105, 109, 141, 145, 151, 215
		<code>\TY@ratio</code>	187,	Z	
		<code>\TY@save@row</code>	105, 109, 110	<code>\Z</code>	5, 6, 347
		<code>\TY@subwidth</code> .	57, 64, 71, 75, <u>93</u> , 413, 417	<code>\z@</code>	10, 21, 36, 63, 64, 81, 108, 123, 148, 165, 169, 186, 197, 201, 235, 239, 247, 258, 273, 276, 307, 318, 325, 374, 377, 380, 382, 384, 386, 407, 409, 410
		<code>\TY@tab</code>	80, 103	<code>\z@skip</code>	225
		<code>\TY@tabarray</code>	88, <u>89</u>		
		<code>\TY@tablewidth</code> 20, 36, 155, 182, 186, 187, 190, 199, 212, <u>315</u>		
		<code>\TY@tabular</code>	15, <u>16</u>		

Change History

v0.1	added	1
General: Initial version	v0.4	
v0.10	<code>\TY@checkmin: \global</code> added	10
General: support <code>\cellcolor</code> see	<code>\xdef</code> not <code>\edef</code>	10
http://tex.stackexchange.com/a/185851/	<code>\TY@tabular</code> : Locally preserve	
global commands		5
v0.11	v0.5	
General: Handle p and b columns	General: Further SPQR	
like m gh/38	modifications to multi pass	
Restore LaTeX counters after	<code>table env</code>	1
trial typesetting gh/12	v0.6	
Update to match latest array	General: Remove multi pass table	
package (rollback to v0.10 for	env and unboxed star form	1
older releases)	<code>\TX@warn</code> : macro added	15
use <code>\hskip1sp</code> to match array	<code>\TY@arrayrule</code> : macro added	6
2.3i from 1996	<code>\TY@classvi</code> : macro added	7
v0.2	v0.7	
General: Changed everything	<code>\TY@tabarray</code> : new macro to	
except the name	support [t] optional arg	7
v0.3	v0.8	
General: Changed everything	General: Rename S to J and ‘hide’	
except the name: s and CLRS	s (until it works)	1