Easy Table

Khanh Ha

Introduction

Easy Table (EZ) is an application tabular package designed to run independently of computer platforms. Its goals are to meet the most rigid requirements from trade typesetters when it comes to tabular work. EZ’s refinements include: precision in row and column, vertical and horizontal placements; baseline and style control for horizontal rules; spanner headings; subspanner headings in four-level nestings; and a table splitting operation.

EZ is a template-controlled program. It requires that table specifications be filled out once, and more important, it allows typing multi-line entries naturally anywhere in a table including in the spanner units. And table rows can be ended gracefully even with gutter rules but without the need to advance to the table’s last column using \&s.

Though a large program with about 200 commands, EZ is easy to learn, set up, and modify. Above all, it is comprehensive and highly precise.

If you master EZ, setting tables will be a joy, not a jolt. And when you reread Knuth’s line, “Printers charge extra when you ask them to typeset tables, …” (The \TeX\book, p. 231), you might have a laugh or two.

Table initialization with \tabinit

Because EZ relies on templates, it requires a proper table format specification before typesetting. Once a table format template is constructed, it is stored and can be reused later on. The \tabinit command is used to build a template for a table. This command specifies the following values:

1. Total columns
2. Table leading
3. Gutter width
4. Preambles

The template’s general usage is:

\tabinit{(total columns)}{(tab leading)}{(gutter width)}{(preambles)}

Consider Table 1:

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Top 10 Newsstand Sellers (000s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Magazine</td>
<td>1982</td>
</tr>
<tr>
<td>_______</td>
<td>______</td>
</tr>
<tr>
<td>TV Guide</td>
<td>9,732</td>
</tr>
<tr>
<td>Family Circle</td>
<td>7,234</td>
</tr>
<tr>
<td>Woman’s Day</td>
<td>9,732</td>
</tr>
<tr>
<td>National Enquirer</td>
<td>5,732</td>
</tr>
<tr>
<td>The Star</td>
<td>9,732</td>
</tr>
<tr>
<td>Penthouse</td>
<td>8,436</td>
</tr>
<tr>
<td>Cosmopolitan</td>
<td>7,795</td>
</tr>
<tr>
<td>Good Housekeeping</td>
<td>5,345</td>
</tr>
<tr>
<td>People Weekly</td>
<td>7,322</td>
</tr>
<tr>
<td>Globe</td>
<td>8,872</td>
</tr>
</tbody>
</table>

CODES:

\tabinit{4}{10pt}{12~t}{\og{.5pt} \og states outside gutter style
\ninepoint
\tttitle{1}{Top 10 Newsstand Sellers (000s)}
\toprul
\hgstub{2em}
\tab{\bf Magazine\&\bf 1982\&\bf 1986\&\bf \% Change\et}
\tab{\en 1.\en TV Guide\&9,732\&8,234\&-13.2\et}
\tab{\en 2.\en Family Circle\&7,234\&6,243\&-15.4\et}
\tab{\en 3.\en Woman’s Day\&9,732\&6,334\&-11.2\et}
\tab{\en 4.\en National Enquirer\&5,732\&8,897\&-23.2\et}
<table>
<thead>
<tr>
<th>Magazine</th>
<th>Circulation</th>
<th>Revenue</th>
<th>Profit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stark</td>
<td>9,732</td>
<td>8,338</td>
<td>-12.3</td>
</tr>
<tr>
<td>Penthouse</td>
<td>8,436</td>
<td>4,039</td>
<td>-43.4</td>
</tr>
<tr>
<td>Cosmopolitan</td>
<td>7,795</td>
<td>5,237</td>
<td>-22.5</td>
</tr>
<tr>
<td>Good Housekeeping</td>
<td>5,395</td>
<td>8,657</td>
<td>-16.7</td>
</tr>
<tr>
<td>People Weekly</td>
<td>7,322</td>
<td>7,342</td>
<td>-14.3</td>
</tr>
<tr>
<td>Globe</td>
<td>8,872</td>
<td>8,764</td>
<td>-11.7</td>
</tr>
</tbody>
</table>

\et

\hrul{3pt}{0} % hrule clears outside rules
\hrul{3pt}{0} % 4-column body span
\hrul{6pt}{1} % hrule joins outside rules


 Calls to \og are of the form:
\og{(dimen)}
The left outside gutter can be either a plain or ruled gutter. Use \og(0pt) if plain; if ruled, fill in units of measure for the rule weights by using one of the five different rule styles below:
- 0.5pt = half point
- 1pt = 1 point
- 2pt = 2 points
- 3pt = 3 points
- 9pt = double rules

 Calls to \C are of the form:
\C{(col. sequence)}{(col. width)}{(gutter style)}
\C{(col. sequence)} is the ordinal number of the column, from left to right. \C{(col. justification)} can have one of three values to determine a column’s paragraph shape: j = justified left/right, l = flush left/ragged right, r = flush right/ragged left, and c = ragged center. \C{(gutter style)} is determined with values as discussed for \og. If the style is 0pt, the gutter is blank; otherwise it has a vrule centered in it.

 Setting table entries with \tab

EZ builds a table by stacking its entries. It sets one entry after another with a command called \tab, ends an entry with another command called \et, and separates the columns in each tab field by the conventional &. Thus:
\tab{(entry)&...&entry}\et

The only command worth discussing here is \et.

 Ending a \tab entry with \et

The purpose of \et is to replace the primitive \cr while running EZ. \et ensures that all gutter rules in effect are output regardless of where the \et is keyed. Needless to say, this helps reduce keystrokes in a multi-column table since stopping short to exit a row no longer requires &s to advance to the end of a row. (Even if a table has no gutter rules, \cr is still unusable. As long as you rely on the template \tabinit you must use \et; bad output will result if you use \cr.)

 The table hrules

Horizontal rules are vital attributes in tables. Their role is to accent the main components by creating demarcations with their style and weight. Before setting a regular hrule three facts need to be determined: the rule’s weight, leading, and style. A complete \hrul command has two parameters:
\hrul{(leading)}{(style)}
\hrul{(leading)}{(style)} is the distance to the hrule’s baseline measured from the base of the line above it; \hrul{(style)} is the way in which the hrule is drawn in relation to possible outside gutter rules. The value of \hrul{(style)} is an integer: a “0” will clear the hrule from outside rules; a “1” will connect the hrule with outside rules.

 Vertical alignment

Tables must cope with three different vertical alignments: top, bottom, and center. Alignments may be changed at any time by stating one of the commands \aligntop, \alignbot, \aligncen.

 Horizontal alignment

Each column style in EZ, justified or ragged, is determined by the \C command in the \tabinit template. Once this style is set, it becomes easier later on to concentrate on the data being entered. However, a need for a change in paragraph style while in a column necessitates a means to control
the paragraph shapes. To change the predetermined paragraph style while in a column one uses: \RR=ragged right; \RL=ragged left; \RC=ragged center; \XR=ragged cancelled (justified) within the appropriate field of a \tab.

Vruled table

Tables with gutter rules should allow two features: 1) All gutter rules in effect should be drawn even in cases where rows end short; 2) Select gutter rules can temporarily be blank for special purposes. Of these two requirements the first can be solved with the use of the command \et.

The \et operation. As previously discussed, the command \et is used to end a \tab row in the same fashion as the \cr. Fortunately, the \et is much more useful than \cr in the face of vrules. In fact, when vrules are present they will be all drawn automatically no matter where you end your row if and only if you use \et.

To learn about the usage of \et we have two illustrations below. First with rows completely filled with data in Table 2:

<table>
<thead>
<tr>
<th>TABLE 2</th>
<th>The use of \et in normal situation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Station</td>
<td>Depth 10m</td>
</tr>
<tr>
<td>3</td>
<td>0.73</td>
</tr>
<tr>
<td>4</td>
<td>0.46</td>
</tr>
<tr>
<td>5</td>
<td>0.78</td>
</tr>
<tr>
<td>6</td>
<td>0.89</td>
</tr>
</tbody>
</table>

Codes:
\ninepoint
\alignncen
\tabinit{8}{10pt}{12pt}{\nog{.5pt}}
\C{1}{5.5pc}{c}{.5pt}\C{2}{26pt}{c}{.5pt}\C{3}{26pt}{c}{.5pt}
\C{4}{26pt}{c}{.5pt}\C{5}{26pt}{c}{.5pt}\C{6}{26pt}{c}{.5pt}
\C{7}{26pt}{c}{.5pt}\C{8}{26pt}{c}{.5pt}
\tttitle{1}{The use of \et in normal situation}
\toprul
\tab{RL}\bf Depth\nl
\RR\bf Station &
\bf 10m & \bf 25m & \bf 75m & \bf 100m & \bf 125m & \bf 150m & \bf 200m \et
\hrul{3pt}{11}
\tab{3&0.73&0.76&0.37&0.08&0.02&0.06&-0.58\et}
\tab{4&0.46&0.45&0.55&0.09&0.13&0.36&-0.76\et}
\tab{5&0.78&0.43&0.67&0.11&0.21&0.08&-0.45\et}
\tab{6&0.89&0.21&0.53&0.42&0.12&0.07&-0.15\et}
\hrul{3pt}{11}

Next with rows ending short in Table 3:

<table>
<thead>
<tr>
<th>TABLE 3</th>
<th>The use of \et in special situation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Station</td>
<td>Depth 10m</td>
</tr>
<tr>
<td>3</td>
<td>0.73</td>
</tr>
<tr>
<td>4</td>
<td>0.46</td>
</tr>
<tr>
<td>5</td>
<td>0.78</td>
</tr>
<tr>
<td>6</td>
<td>0.89</td>
</tr>
</tbody>
</table>
Thus far we have seen the usefulness of \&et, especially in multicolumn \vrule ruled tables where one can exit a row gracefully, forgetting all the remaining tab alignments one would have needed to advance to reach the end of the current row. That leaves us with a second feature to explore: how to temporarily empty specific ruled gutters.

**Gutter rules temporarily blank with \bgut.**

In order to void any gutter rules within a table body one must state a *range*, i.e., the starting column and the ending column; this range includes the gutter rules to be blank:

\bgut\{\textcolor{red}{\texttt{\textbackslash start col}}-\textcolor{red}{\texttt{\textbackslash end col}}} \]

where the hyphen (-) stands for possible columns in between, and the pair of brackets form the command’s delimiters. This command can be stated either before or after the \tab starts; its effect will be limited to only the row it is issued for.

Table 4 shows its usage:

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>14</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
<td>30</td>
<td></td>
</tr>
</tbody>
</table>
Spanners

Spanners are tough customers. The first type of spanner is a column spanner, and the second a row spanner. Column spanners are entries that straddle a number of columns; they must be able to wrap around automatically in multi-line paragraph fashion with no contrived manual line-breaking operation; they must also be able to justify within their own territory in terms of raggedness or left and right justification; they ought to relate in vertical alignment to their possible counterparts on the same level in the row; their spanner rules must be flexible enough to remain within their own width or extend to join the neighboring gutter-rules in addition to their rules' vertical adjustability for a particular leading; and they must be able to nest other spanners.

The second type of spanner is the row spanner. This kind is a column which spans vertically a number of rows and serves as their common heading.

**Body spanners with \bspan.** In a table body one might come across an entry that spans a number of columns. This type of column spanner must be specified by a range, i.e., the starting column and the ending column of the spanned columns. **EZ**'s command for the body column spanner is \bspan, which is delimited by a pair of brackets:

\bspan[(start col)-(end col)]

\bspan must be stated inside the \tab command. Once the span is active, data can be poured into this space and **EZ** will handle the line-breaking algorithm in addition to the paragraph shape that has been specified. This shape is controlled by the style in the column that starts the body spanner, which has been specified in the \tabinit template. One can always override this style with one of the following commands: \JUST, \RR, \RL, \RC. Table 5 demonstrates the use of \bspan:

<table>
<thead>
<tr>
<th>SUN</th>
<th>MON</th>
<th>TUES</th>
<th>WED</th>
<th>THUR</th>
<th>FRI</th>
<th>SAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Beat midsummer heat the cool and easy way with delightful, delicious dairy-fresh ice cream. Dip into the natural goodness of America's favorite treat. Build yourself the sundaes of your dreams with your favorite fruits, nuts and syrups, or with the classic whipped cream and a cherry.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>11</td>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td>21</td>
<td>22</td>
</tr>
<tr>
<td>23/30</td>
<td>24/31</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
</tr>
</tbody>
</table>

**CODES:**
\ninepoint
\tabinit{7}{10pt}{12pt}{\log{.5pt}}
\C{1}{3pc}{1}{.5pt}\C{2}{3pc}{1}{.5pt}\C{3}{3pc}{1}{.5pt}\C{4}{3pc}{1}{.5pt}\C{5}{3pc}{1}{.5pt}\C{6}{3pc}{1}{.5pt}\C{7}{3pc}{1}{.5pt}
Beat midsummer heat the cool and easy way with delightful, delicious
dairy-fresh ice cream. Dip into the natural goodness of America's
favorite treat. Build yourself the sundae of your dreams with your
favorite fruits, nuts and syrups, or with the classic whipped cream
and a cherry.

Because a body spanner always straddles a
number of columns, it must collect an equal number of tab alignments from these spanned columns in
order to move past them to the next column. Thus the spanner in this example needs six &s for six
skipped columns. The paragraph shape of this spanner is controlled by the command “1” in the
\tabinit template for the second column which
starts the spanner. Again overriding the text shape
of the spanner is simple, as already discussed.

**Head spanners with \spaninit.** The main difference between a body spanner and a head spanner
is that the first acts like a header crossing a number of designated columns (it has no entries beneath it)
while the second works as a header but also has entries under it and often has an hrule separating
the header from its spanned entries. Therefore a head spanner must be treated as one complete unit
that combines the header, the rule, and the spanned entries. To **EZ** each unit requires a column for
itself; thus, for example, a 3-column spanner will occupy one single column and within this column
the three spanned entries. **EZ**'s method of setting
a row with column spanners is to set each spanner
unit in a column completely, then move onward to
the next possible column. It does not set across line
for line as a typewriter would. Users need not guess
entry depths nor make awkward attempts to break
entries manually before setting the spanners. For
clearly, **EZ** calls the head spanner spanner head and
its spanned entries spanner cells; the separating rule
between these two components is termed a spanner
rule.

When a table has spanner heads, **EZ** uses
this principle: the main body's template must be
designed first before that of the spanner heads.
Thus two templates will be present: 1) \tabinit
for the main body, and 2) \spaninit for the column
headings with spanner units.

**Spanner units in vruled tables.** The design
for spanner units is the same in both table styles,
vruled or plain. Spanner units which are ruled on
all sides normally would incorporate sub-spanner
units because these rules serve as demarcation. In
plain tables the spanner units are single-level.

Now consider Table 6 for the introduction of
spanner units:
TABLE 6  Vruled Spanner: Single Level

<table>
<thead>
<tr>
<th>Size °</th>
<th>Amps</th>
<th>Reserve capacity</th>
<th>Battery case size, inches</th>
<th>Catalog Number</th>
<th>Wt. lbs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>24°</td>
<td>500</td>
<td>115</td>
<td>10 L 6 W 8 H</td>
<td>28K 4315N</td>
<td>41</td>
</tr>
<tr>
<td>34°</td>
<td>502</td>
<td>116</td>
<td>20 L 7 W 8 H</td>
<td>28K 4314N</td>
<td>42</td>
</tr>
<tr>
<td>44°</td>
<td>504</td>
<td>117</td>
<td>30 L 8 W 8 H</td>
<td>28K 4313N</td>
<td>43</td>
</tr>
<tr>
<td>54°</td>
<td>506</td>
<td>118</td>
<td>40 L 9 W 8 H</td>
<td>28K 4312N</td>
<td>44</td>
</tr>
<tr>
<td>64°</td>
<td>508</td>
<td>119</td>
<td>50 L 2 W 8 H</td>
<td>28K 4311N</td>
<td>45</td>
</tr>
<tr>
<td>74°</td>
<td>510</td>
<td>120</td>
<td>60 L 3 W 8 H</td>
<td>28K 4319N</td>
<td>46</td>
</tr>
</tbody>
</table>

This table shows two templates, one for the main body, another for the column headings with spanner units. As already mentioned, the main body’s template must be given first; for this table it prepares 8 columns. Next the template for the headings is designed for 6 columns because of the 3-column spanner unit—this unit occupies one column. We turn now to the method of computing the total width of the spanner head.

**Setting the spanner heads with** `\main`. The command for spanner head is `\main`. Its form is: `\main{(Spanner Head Text)}{(Spanner Rule Style)}`
where the spanner head text in the first argument can be multi-line data whose line-breaking algorithm is determined by the paragraph style commands, i.e., justified or ragged. The spanner rule style in the second argument controls the length of the rule: if “1”, it will join the adjacent gutter rules; if “0”, it will remain within the true width (the width of the spanner head).

**Setting the spanner cells with \main.** Spanner cells are grouped together under the main spanner head. The adjective main is used to distinguish the first-level spanner head from the second-, third-, and fourth-level spanner heads. This complication arises when the column headings incorporate subspanner units inside the main one. EZ allows four levels of spanners, descending from main to sub to subsub to subsubsub spanners. Regardless of the possible existence of the sub spanners, all entities under the main spanner head must start with \startmain and end with \endmain:

\begin{verbatim}
\main{{spanner head}}{{rule style}}
\startmain
\maine{{dimen}}{{cell text}}
\endmain
\end{verbatim}

where \maine is a command for each spanner cell that looks for two arguments: 1) width of spanner cell, 2) text of spanner cell. The I will draw a gutter rule whose default weight is .5pt.

**Setting the spanner rules.** A spanner rule can have one of three styles specified by: “0” to stay within the true width of the spanner head; “1” to extend the rule to join adjacent, but not outside, gutter rules; “2” to extend rule to intersect all adjacent, including outside, gutter rules. Besides this option, the rule leading default (3pts) can be changed any time by stating \sprskip{(dimen)}

where \sprskip is the leading amount for the spanner rule. The \sprskip command affects the leading change for the spanner rule only in the current column; thus changing leading for multiple spanner rules in different columns requires the \sprskip command to be restated each time in a new column. Making two changes to the rule leading and style of the previous example, we have:

\begin{verbatim}
\ninepoint
\restorespani6) % most recent spanner template
\toprul
\tab{Size&Amps&Reserve capacity& Battery case size, inches& Catalog Number& Wt. lbs.}
\sprskip(6pt) % rule leading is now 6pt (good for one column only)
\main{Battery case size, inches}{0} % Style 0: not touching gutter rules
\startmain
\maine{1pc}{L} | \maine{1pc}{W} \maine{1pc}{H}
\endmain
\et}
\hrul(3pt)(0)
\end{verbatim}

**Multiple spanners.** When a row has more than one spanner unit it must resolve an issue of alignment for the spanner heads and spanner cells. All the spanner heads on a row will be related in their vertical alignment, and so are the spanner cells. Our first look at multiple spanners focuses on Table 7:
<table>
<thead>
<tr>
<th>Months and year</th>
<th>Number of stoppages</th>
<th>Workers involved</th>
<th>Days idle</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Beginning in month or year</td>
<td>In effect during month</td>
<td>Beginning in month or year (in thousands)</td>
</tr>
<tr>
<td>1947</td>
<td>270</td>
<td>1,629</td>
<td>25,720</td>
</tr>
<tr>
<td>1948</td>
<td>245</td>
<td>1,435</td>
<td>26,127</td>
</tr>
<tr>
<td>1949</td>
<td>262</td>
<td>2,537</td>
<td>43,420</td>
</tr>
<tr>
<td>1950</td>
<td>424</td>
<td>1,698</td>
<td>30,390</td>
</tr>
</tbody>
</table>

CODES:
\alignncen
\ninepoint
\tabinit{7}{}{6pt}{\og{.5pt}} % 7 columns for body
\c{1}{8pc}{1}{2pt}\c{2}{3.5pc}{c}{.5pt}\c{3}{3.5pc}{c}{2pt}\c{4}{3.5pc}{c}{.5pt}
\c{5}{3.5pc}{c}{2pt}\c{6}{3.5pc}{c}{.5pt}\c{7}{3.5pc}{c}{.5pt}
\spaninit{4}{}{6pt}{\og{.5pt}} % 4 columns for headings
\c{1}{8pc}{1}{2pt}\c{2}{7.5pc}{c}{2pt}
\c{3}{7.5pc}{c}{2pt}\c{4}{7.5pc}{c}{.5pt}
\toprule
\title{Multiple Spanners}
\toprule
\tab{Months and year#}
\main{Number of stoppages}{1}
\startmain
\main{3.5pc}{Beginning in month or year} &
\main{3.5pc}{In effect during month}
\endmain
&
\main{Workers involved}{1}
\startmain
\main{3.5pc}{Beginning in month or year (in thousands)} &
\main{3.5pc}{In effect during month (in thousands)}
\endmain
&
\main{Days idle}{2} % Style 2: intersecting outside rule
\startmain
\main{3.5pc}{Number (in thousands)} &
\main{3.5pc}{Percent of estimated working time}
\endmain
\et}
\hrul{6pt}{1}
\tab{1947\dotlead&270\dotlead&1,629\dotlead&25,720\dotlead&}\et}
\tab{1948\dotlead&245\dotlead&1,435\dotlead&26,127\dotlead&.22}\et}
\tab{1949\dotlead&262\dotlead&2,537\dotlead&43,420\dotlead&.38}\et}
\tab{1950\dotlead&424\dotlead&1,698\dotlead&30,390\dotlead&.26}\et}
Notice the correct vertical alignment of each component in all the spanner units. The spanner heads and also the spanner cells fulfill their assignments once they are able to relate vertically to each other. To get a better look at this vertical relation, examine Table 8:

<table>
<thead>
<tr>
<th>Warp No.</th>
<th>Fabric Description</th>
<th>Thickness (in.)</th>
<th>Weight (oz/yd²)</th>
<th>Tensile Strength (lb/in.)</th>
<th>Tear Strength (lb)</th>
<th>Peel Strength (lb)</th>
<th>Flexural Rigidity* (lb/in.$^2$/in.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

CODES:
\hrul{6pt}{1}

\begin{tabular}{|l|l|l|l|l|l|}
\hline
Warp No. & Fabric Description & Thickness (in.) & Weight (oz/yd²) & Tensile Strength (lb/in.) & Tear Strength (lb) & Peel Strength (lb) & Flexural Rigidity* (lb/in.$^2$/in.) \\
\hline
\end{tabular}

\hrul{6pt}{1}

\begin{tabular}{|l|l|l|l|l|l|}
\hline
| Warp & Fill | Warp & Fill | Warp & Fill | Warp & Fill |
|-------|--------|--------|--------|--------|--------|--------|
\hline
|          |         |         |         |         |         |         |
\hline
|          |         |         |         |         |         |         |
\hline
|          |         |         |         |         |         |         |
\hline
\end{tabular}
Nested spanners. Nested spanners occur when a spanner unit encloses another spanner unit which, in turn, comprises another one and so on. EZ supports four levels of nested spanners descending from the main to sub to subsub to subsubsub spanners. The commands for nested spanners are: \texttt{\textbackslash sub}, \texttt{\textbackslash ssb}, and \texttt{\textbackslash sssub}, all below the main level of \texttt{\textbackslash main} and restricted within the limit marked by \texttt{\textbackslash startmain} and \texttt{\textbackslash endmain}. All the subspanner units have the following usage:

\texttt{\textbackslash sub\{subspanner heading\}\{rule style\}\{subcell entries\}}
\texttt{\sub\{ssubspanner heading\}\{rule style\}\{ssubcell entries\}}
\texttt{\ssub\{sssubspanner heading\}\{rule style\}\{sssubcell entries\}}

where the first two arguments are like those in \texttt{\textbackslash main}. Only the third argument merits attention because this extra argument makes \texttt{\textbackslash sub} (and its counterparts) different from \texttt{\textbackslash main}. A complete look at a subspanner unit:

\texttt{\textbackslash sub\{subspanner heading\}\{rule style\}\{subcell entries\}}
\texttt{\sub\{cell width\}\{entry text\} |}
\texttt{\sub\{cell width\}\{entry text\} |}
\texttt{\sub\{cell width\}\{entry text\} |}
\texttt{\sub\{cell width\}\{entry text\} |}
\texttt{\sub\{cell width\}\{entry text\} |}
\texttt{\sub\{cell width\}\{entry text\} |}
\texttt{\sub\{cell width\}\{entry text\} |}

where \texttt{\textbackslash sub} is much like \texttt{\textbackslash main} and \texttt{\textbackslash sub} is the dividing rule between subspanner cells. The focus here is on the third argument for each subspanner unit: it must have matching braces to avoid ”Runaway argument” errors. Spaces after the argument’s open brace and also spaces before its close brace are ignored. To see how nested spanners work, examine Table 9:

<table>
<thead>
<tr>
<th>County</th>
<th>Industry</th>
<th>Employment change</th>
<th>Effect of shift-share by</th>
<th>Location quotient</th>
</tr>
</thead>
<tbody>
<tr>
<td>Santa Cruz (Arizona)</td>
<td>Apparel..</td>
<td>Actual 7</td>
<td>28</td>
<td>48</td>
</tr>
</tbody>
</table>

CODES:

\texttt{\ninepoint}
\texttt{\aligncen}
\texttt{\tabinit7\{10pt\}\{6pt\}\{log.5pt\}} \% 7 cols for body
\texttt{\C1\{4pc\}\{1.5pt\}\C2\{4pc\}\{1.2pt\}\C3\{3pc\}\{c\}{.5pt}\C4\{3pc\}\{c\}{.5pt}}
\texttt{\C5\{3pc\}\{c\}{.5pt}\C6\{3pc\}\{c\}{2pt}\C7\{3pc\}\{c\}{.5pt}}

\texttt{\spaninit4\{10pt\}\{6pt\}\{log.5pt\}} \% 4 cols for headings
\texttt{\C1\{4pc\}\{1.5pt\}\C2\{4pc\}\{1.2pt\}\C3\{13.6pc\}\{c\}{2pt}\C4\{3pc\}\{c\}{.5pt}}

\texttt{\topru1}
\texttt{\title1\{Nested Spanners\}}
\texttt{\topru1}

\texttt{\tab\{County\& Industry\&}
\texttt{\main\{Employment change\}\{1\}}
\texttt{\startmain}
\texttt{\main\{3pc\}\{Actual\} |}
\texttt{\sub\{Effect of shift-share by\}\{1\} \% subspanner cells go in here}
More nested spanners

When spanners start to nest in multilevel fashion, the user is responsible for keeping track of each one, its start and its end. In practice it helps to type empty brace pairs first to lock up the design in question, then start pouring in text data. It also helps to use a text editor which has a brace-checking utility to spot unbalanced braces. Table 10 features spanners with three levels deep:

<table>
<thead>
<tr>
<th>TABLE 10 Nested Spanners</th>
</tr>
</thead>
<tbody>
<tr>
<td>Year</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>Year</td>
</tr>
</tbody>
</table>

CODES:
\fivepoint
\aligncen
\tabinit{13}{7pt}{.5pt}{\og{Opt}}
\C(1){1.5pc}(1){2pt}\C(2){3pc}(c){2pt}\C(3){3pc}(c){.5pt}\C(4){3pc}(c){.5pt}
\C(5){1.5pc}(c){1.5pt}\C(6){3pc}(c){.5pt}\C(7){3pc}(c){.5pt}\C(8){1.6pc}(c){.5pt}
\C(9){3pc}(c){.5pt}\C(10){3pc}(c){.5pt}\C(11){3pc}(c){.5pt}
\C(12){3pc}(c){2pt}\C(13){3pc}(c){0pt}
\spaninit{4}{7pt}{.5pt}{\og{Opt}}
\C(1){1.5pc}(1){2pt}\C(2){3pc}(c){2pt}\C(3){28.5pc}(c){2pt}\C(4){3pc}(c){0pt}
Spanners forever. We close this section about column spanners with an example whose details in drawing nested spanners should be studied carefully by readers to gain an idea about EZ's principles in spanner design:

**TABLE 11** Heavy-Duty Spanners

<table>
<thead>
<tr>
<th>Company and Market</th>
<th>Price</th>
<th>Volume</th>
<th>Earnings and Dividends</th>
<th>P/E Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Last Week's Close</td>
<td>Last Week's</td>
<td>Last Year's</td>
<td>Shares Outstanding</td>
</tr>
<tr>
<td></td>
<td>Last Week</td>
<td>4 Wks.</td>
<td>Year to Date</td>
<td>High</td>
</tr>
<tr>
<td>BaltGas</td>
<td>$14.88</td>
<td>0</td>
<td>2.6</td>
<td>$23.88</td>
</tr>
<tr>
<td></td>
<td>%</td>
<td>%</td>
<td>%</td>
<td>$</td>
</tr>
</tbody>
</table>

**CODES:**
```
\vskip-2pt
\hrul
\endmain
\not in labor force\et
&Not in labor force\et
\vskip-2pt
\hrul
\endmain
\&Not in labor force

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|}
\hline
Company & Price & Volume & Earnings & P/E Ratio \\
and Market & Last Week's Close & Last Week's & Last Year's & Shares Outstanding & Last 12 Months & 5 Yr. Annual Growth Rate & Dividend Yield & P/E Ratio \\
\hline
\hline
\end{table}
```

\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|}
\hline
\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|}
\hline
\end{tabular}
```
```
Row spanners. In contrast to column spanners which cross columns, row spanners cross rows. The cell supposedly acting as a row spanner will vertically straddle a number of rows; in EZ such a spanner is achieved with \xrow. Its usage is:

\row\{(entry text)\}

and it is treated like a normal column. Table 12 provides an example:
**Table 12: Row Spanners**

<table>
<thead>
<tr>
<th>BATTERIES NOT INCLUDED. FULL 1-YEAR MONEY BACK GUARANTEE</th>
<th>Subtotal $</th>
</tr>
</thead>
<tbody>
<tr>
<td>NY residents add sales tax $</td>
<td></td>
</tr>
<tr>
<td>Add $2 shipping no matter how many you order $</td>
<td></td>
</tr>
<tr>
<td><strong>TOTAL $</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Codes:**

```
\ninpoint
\ntabinit{3}{10pt}{6pt}{\og{.5pt}}
  \C{1}{7.5pc}{c}{.5pt}\C{2}{11.5pc}{r}{.5pt}\C{3}{3pc}{l}{.5pt}}
\n\tttitle{1}{Row Spanners}
\ntoprul
\ntab{\xrow{bf BATTERIES NOT INCLUDED. FULL 1-YEAR MONEY BACK GUARANTEE}
  & Subtotal & $}
  \prul[2-3,6pt] % partial rule for 2-3 cols; on 6pt leading
  \blank{2pt}
  \tab{\& NY residents add sales tax & $}
  \prul[2-3,6pt]
  \blank{2pt}
  \tab{\& Add $2 shipping no matter how many you order & $}
  \hrul{6pt}{1}
  \blank{2pt}
  \tab{\& RL TOTAL & $}
  \hrul{6pt}{1}
```

Note that the row spanner acts like a normal column because, after setting it, one must tab across to the next column. While working with a row spanner, \prul will be needed to draw partial hrules which avoid the cell the row spanner occupies. Once beyond the row spanner, the normal \hrul for full-width hrules can be used.

**Conclusion.** This article is excerpted from the 118-page operation manual of Easy Table. Its purpose is to illustrate a few major features of this software. For more information about the purchase of Easy Table software, please contact me at 301-598-0557, or write to:

○ Khanh Ha  
  14912 Village Gate Drive  
  Silver Spring, MD 20906