"Hey — it works!"

Jeremy Gibbons
University of Auckland
jeremy@cs.auckland.ac.nz

Welcome once again to *"Hey — it works!"*, a column devoted to tips and tricks for (LA)TEX. I am always on the lookout for interesting such articles (especially short ones). If you have solved a (LA)TEX problem in a novel way, please write it up and send it in! No matter how trivial it seems to you, the chances are that it will be just what someone else needs.

In this issue we have two contributed pieces, and three responses to an item of mine from the last issue. The contributed pieces are from Michael Barr, on under- and over-brackets and -parentheses similar to the under- and over-braces in plain TEX, and from John Langer, on acronyms that automatically explain themselves on first use.

We start with the responses generated by my item in the last issue, on the macro that I use for generating the subsection headings in this column. Evidently this item was not as elegant as it might have been! Just for reference, here is my definition:

```
\def\squashedsubsection#1#2{%
   \subsection*{%
      \hbox to \linewidth{%           a box as wide as the page
         #1%                          subsection title
         \hfil%                       stretchy space
         \llap{\normalsize%           ignore width of tabular
            \begin{tabular}[t]{r@{}}
               #2%                    author's name and address
            \end{tabular}}}}}
```

Squashed heads

Kees van der Laan
Hunzeweg 57, 9893PB
Garnwerd, The Netherlands
cgl@rc.service.rug.nl

Kees van der Laan suggested an approach based on Knuth's `\beginchapter` macro from Appendix E of the TEXbook. He separates the author's name, affiliation and email address into three arguments, and defines (in effect)

```
\def\squashedsubsection#1#2#3#4{%
   \hbox to \linewidth{{\large\bf#1}\hfil#2}\let\\\cr
   \halign{\hbox to \linewidth{\hfil##}\cr#3\\\tt#4\\}}
```

The title above was generated by

```
\squashedsubsection{Squashed heads}{Kees van der Laan}
   {Hunzeweg 57, 9893PB\\Garnwerd, The Netherlands}
   {cgl@rc.service.rug.nl}
```

This is a better solution to the problem of addresses in headings

Michael Barr
McGill University
barr@math.mcgill.ca

Michael Barr observed that my approach did not allow the title to extend to more than one line. He suggested, in effect, placing argument `#1` in a tabular, just as I did with `#2`; thus, the line containing `#1` in my definition should be replaced by

```
\rlap{%
   \begin{tabular}[t]{@{}l}
      #1%              subsection title
   \end{tabular}}%
```

Then the first argument can contain occurrences of `\\`. The above title was generated by

```
\squashedsubsection{This is a better solution to the problem of\\
    addresses in headings}
   {Michael Barr\\McGill University\\\tt barr@math.mcgill.ca}
```

It is not clear whether the author's name and address is better aligned with the *first* or the *last* line of the title; the latter takes more space, but avoids having different baselines on the left and right of the page. This latter behaviour can be obtained by replacing the `[t]` within the `\rlap` by `[b]`.

This is a better solution to the problem of addresses in headings, linebreaking the title automatically

Mike Piff
University of Sheffield
M.Piff@shef.ac.uk

Mike Piff suggested a third way, based on Knuth's `\signed` macro from Chapter 14 of the TEXbook. He defines

```
\long\def\squashedsubsection#1#2{%
   \subsection*{%
      \raggedright
      #1%
```

```
      \unskip\nobreak\hfill\penalty50 \hskip1em
      \mbox{}\nobreak\hfill
      {\normalsize
      \begin{tabular}[t]{r@{}}#2\end{tabular}}%
      \parfillskip=0pt \finalhyphendemerits=0}}
```

This has the advantage of letting TEX make the line breaks in the title, so the first argument need have no \\s (but on the other hand, it does not allow the author's name to be aligned with the *first* line of the title). The above title was generated by

```
\squashedsubsection{This is a better solution to the problem of
   addresses in headings, linebreaking the title automatically}
   {Mike Piff\\University of Sheffield\\\tt M.Piff@shef.ac.uk}
```

With this solution, enough space is left for the whole width of the author's address, even if the first line (the author's name) is relatively short. An alternative might be to give the name as an argument by itself, defining

```
\long\def\squashedsubsection#1#2#3{%
   \subsection*{%
      \raggedright
      #1%
      \unskip\nobreak\hfill\penalty50 \hskip1em
      \mbox{}\nobreak\hfill
      {\normalsize
      \begin{tabular}[t]{r@{}}
         #2\\\llap{\begin{tabular}{r@{}}#3\end{tabular}}
      \end{tabular}}%
      \parfillskip=0pt \finalhyphendemerits=0}}
```

This produces, for example,

This is a better solution to the problem of addresses in headings, linebreaking the title automatically

Mike Piff
University of Sheffield
`M.Piff@shef.ac.uk`

Are there any more solutions to the problem out there?

Under- and over-brackets and parentheses

Michael Barr
McGill University, Canada
barr@triples.Math.McGill.CA

A couple years ago, a question was posted on `comp.text.tex` about how to make under- and over- square brackets and parentheses similar to under and over braces. It proved quite easy to modify those macros; in one way, it was easier, since you do not need center pieces. The curly parts for the braces can be used for the parentheses, but you have to manufacture with rules the end pieces for the brackets, since they are not provided in the standard fonts. Here is the result, in a form that I have tested in both LATEX and plain TEX. The results look a bit thick, but are exactly the same thickness as with the braces.

The code

```
$$ \overbrack{x+\cdots+x}^{k\rm\;times} \qquad
   \underbrack{x+y+z}_{>\,0}            \qquad
   \overparen{x+\cdots+x}^{k\rm\;times} \qquad
   \underparen{x+y+z}_{>\,0}$$
```

generates the following output:

$$\overbracket{x+\cdots+x}^{k\text{ times}} \qquad \underbracket{x+y+z}_{>\,0} \qquad \overparen{x+\cdots+x}^{k\text{ times}} \qquad \underparen{x+y+z}_{>\,0}$$

```
\catcode`\@=11
\def\downparenfill{$\m@th\braceld\leaders\vrule\hfill\bracerd$}
\def\upparenfill{$\m@th\bracelu\leaders\vrule\hfill\braceru$}
\def\overparen#1{\mathop{\vbox{\ialign{##\crcr\noalign{\kern3\p@}
      \downparenfill\crcr\noalign{\kern3\p@\nointerlineskip}
      $\hfil\displaystyle{#1}\hfil$\crcr}}}\limits}
\def\underparen#1{\mathop{\vtop{\ialign{##\crcr
      $\hfil\displaystyle{#1}\hfil$\crcr
      \noalign{\kern3\p@\nointerlineskip}
      \upparenfill\crcr\noalign{\kern3\p@}}}}\limits}

\newbox\brackd \newbox\bracku
\setbox\brackd=\hbox {\vrule height 0pt depth
  3pt width 1pt}\ht\brackd=0pt\dp\brackd=1pt
\setbox\bracku=\hbox {\vrule height 3pt depth
  0pt width 1pt}\ht\bracku=1pt\dp\bracku=0pt
\def\downbrackfill
   {$\m@th\copy\brackd\leaders\vrule\hfill\copy\brackd$}
\def\upbrackfill
   {$\m@th\copy\bracku\leaders\vrule\hfill\copy\bracku$}
```

```
\def\overbrack#1{\mathop{\vbox{\ialign{##\crcr\noalign{\kern3\p@}
      \downbrackfill\crcr\noalign{\kern3\p@\nointerlineskip}
      $\hfil\displaystyle{\kern1pt#1\kern1pt}\hfil$\crcr}}}\limits}
\def\underbrack#1{\mathop{\vtop{\ialign{##\crcr
      $\hfil\displaystyle{\kern1pt#1\kern1pt}\hfil$\crcr\noalign
     {\kern3\p@\nointerlineskip}
      \upbrackfill\crcr\noalign{\kern3\p@}}}}\limits}
\catcode`\@=12
```

Self-expanding acronyms

John V. Langer
The Aerospace Corporation, LA
langer@aero.org

We compose technical documents that employ a large number of acronyms. It is considered good form to explain the meaning of a given acronym the first time it is used, then use the acronym without explanation thereafter. With the marvels of electronic editing, paragraphs and sections get moved around quickly making it easy to inadvertently have an acronym employed before its definition is provided. A secondary hazard is that one may, by accident or habit, wind up explaining the acronym every time it is used, causing considerable annoyance to the reader. To get around this, we came up with the following very simple LaTeX macro, best illustrated via example:

```
\newcommand{\gps}
   {Global Positioning System (GPS)\renewcommand{\gps}{GPS}}
```

This creates a macro called `\gps` that will produce the lengthy explanation the first time it is used and the shorter acronym in every subsequent use. For example,

```
The \gps\ system is an extension of concepts first investigated
in the late 1960's, when \ldots.  By late 1979, four Block I
\gps\ satellites had been deployed, and in 1993 a full
\gps\ constellation was deployed.
```

produces

> The Global Positioning System (GPS) system is an extension of concepts first investigated in the late 1960's, when . . . . By late 1979, four Block I GPS satellites had been deployed, and in 1993 a full GPS constellation was deployed.