

## Glisterings

Peter Wilson

The aim of this column is to provide odd hints or small pieces of code that might help in solving a problem or two while hopefully not making things worse through any errors of mine.

We'll meet again,  
Don't know where, don't know when,  
But I know we'll meet again  
Some sunny day.

---

*We'll Meet Again,*  
ROSS PARKER & HUGHIE CHARLES

### 1 Repetition

In September 2009 JT posed the following to the `comp.text.tex` newsgroup (`ctt`).

*I have numbered propositions of the form:*

- (P1) Some proposition.
- (P2) Another proposition.
- (P3) Yet another proposition.

*which are in a custom list environment and I can refer to the labels (P1, P2, etc) later in the document. However I sometimes want to also repeat the corresponding proposition like this:*

Recall P2 from Chapter 1:

- (P2) Another proposition.

*Is there any way to output the entire list item without having to retype it?*

Lars Madsen [5] responded with the following example code.

```
\documentclass[a4paper]{memoir}
\makeatletter
\newcommand{\Reuse}[1]{\@nameuse{forlater@#1}}
\newcommand{\ForLater}[2]{%
  \item[(#1)]\def\@currentlabel{#1}\label{#1}%
  \global\long\@namedef{forlater@#1}{#2}%
  \Reuse{#1}}
\makeatother
\begin{document}
\begin{itemize}
\ForLater{P1}{Some long text}
\ForLater{P2}{More longer text.\par
  In paragraphs.}
\end{itemize}
```

```
Recall \ref{P2}:
\begin{itemize}
\item[(\ref{P2})] \Reuse{P2}
\end{itemize}
\end{document}
```

As a demonstration that Lars' `\ForLater` and `\Reuse` macros work, I used them in the description above of JT's request.

In an earlier column I had tackled the question of repeating work in a somewhat different, and a not quite so elegant, manner [8].

---

*verbatim et litteratim* — word for word and letter for letter.

*Chambers Dictionary*

## 2 Verbatims

### 2.1 `\verb` with an argument

Luca Merciadri asked on `ctt` if there was a way of defining a `\verb` macro that took the verbatim material as an argument enclosed in braces.

Ulrich Diez [2] responded with three solutions, the last two of which avoided any assignments. The first, shown below, looked much simpler to me.

```
\edef\verba#1#{\noexpand\verb#1\string}%
  \let\noexpand\next=}
```

```
\edef\verba#1#{%
  \noexpand\verb#1\string}%
  \noexpand\expandafter
  \noexpand\fi
  \noexpand\if{\noexpand\iffalse}\noexpand\fi}
```

```
\edef\verba#1#{%
  \noexpand\verb#1\string}%
  \noexpand\expandafter\expandafter
  \noexpand\csgname @gobble\endcsgname
  \noexpand\string}
```

You can use `\verba` like this:

'You can use either `\verba{the \verba macro}` or `\verba*{the \verba* macro}`, whichever suits.'

which will produce:

```
'You can use either the \verba macro or
the \verba* macro, whichever suits.'
```

However, just like `\verb`, `\verba` with its argument, cannot be used in an argument to another macro, not even in the argument to `\verba`.

### 2.2 Automatic line breaking

Hans Balsam asked on `ctt`:

*I'm looking for a way to combine the features of the verbatim environment and L<sup>A</sup>T<sub>E</sub>X's automatic line breaking.*

'Zappathustra' (Paul Isambert) responded [3]:

```
\makeatletter
\def\xobeysp{ }
\makeatother
```

This redefines `\xobeysp`, to which the space character is `\let` in verbatim text, to a normal space instead of an unbreakable space. Then you can use the usual ‘verbatim’ environment.

This proposal works, albeit with at least one surprise — a space following a comma gets swallowed so a double space should be used instead of a single space. The other potential surprises are that hyphenation is disabled and multiline verbatim text is set ragged right.

If any man will draw up his case, and put his name at the foot of the first page, I will give him an immediate reply. Where he compels me to turn over the sheet, he must wait my leisure.

---

*Memoirs*, LORD SANDWICH

### 3 Small pages

Harald Hanche-Olsen asked [slightly edited] on `ctt`: *I’d like to make some PDF files especially for reading on screen, more specifically on the iphone. For much of this, a fixed page length seems like a straitjacket. I want to divide the material into pages so that one topic will fit on one page. Some pages will be very short while others will be very long. I don’t want oceans of white space at the bottom of the pages.*

*I imagine doing this with L<sup>A</sup>T<sub>E</sub>X ... [but] the output routine gives me goose bumps ... I don’t plan on using marginal notes and if I must do without floats and footnotes, that is fine too. I could of course do it in plain T<sub>E</sub>X, but would like to have the added power of L<sup>A</sup>T<sub>E</sub>X available.*

Will Robertson responded [7] with a potential solution based on the `preview` package [4]. His code follows, and I have taken the liberty of extending it very slightly to enable it to work with a variety of classes, and also extending the example document.

```
%\documentclass[article]{memoir}
\documentclass{article}
\usepackage{charter}% more readable on a screen
\usepackage{lipsum}
\makeatletter
%% PW’s extension here
\ifclassloaded{memoir}{%
  \let\section\chapter
  \let\raggedy\raggedyright
}{\usepackage{ragged2e}
  \let\raggedy\RaggedRight}
\makeatother

\usepackage[active,tightpage]{preview}
\usepackage{hyperref}
\newenvironment{page}{%
  \begin{preview}
```

```
\begin{minipage}{5cm}
\medskip \centering
\begin{minipage}{4.5cm}
\footnotesize\raggedy
\parindent=2em
}{%
  \end{minipage}
\medskip
\end{minipage}\end{preview}}

\begin{document}

\begin{page}\tableofcontents\end{page}
\begin{page}
\section{Foo}
\lipsum[1]
\end{page}
\begin{page}
\section{Bar}\lipsum[2-5]\end{page}
\begin{page}
\section{Fuz}
Some text here. I wonder if one can have a
marginal note. %\marginpar{At the side}
It doesn’t work!
\section{Fuzzy}
What if we have two ‘sections’ on the
same page?
\end{page}
\begin{page}
\section{Fie}
Some text here. I wonder if one can have a
footnote.\footnote{At the end like this}
It works!\par
\lipsum[2-5]
\end{page}
\end{document}
```

It is not possible to demonstrate Will’s page environment here, but it does seem to meet Harald’s request as far as I understand it. Floats do not work, and page numbers are not printed, but `\tableofcontents` and the `hyperref` package [6] work (at least as used in the test code above), if needed.

---

He fixed thee ’mid this dance  
Of plastic circumstance.

---

*Rabbi Ben Ezra*, ROBERT BROWNING

### 4 Prefixing section heads

‘Ghoetker’ wrote to `ctt` along the following lines: *I have changed the formatting of subsections in my document to start with the term ‘Activity’ (trust me, it made sense). So, the subsection heading is ‘Activity A.1. Test’. When I cross-reference, however, this isn’t what I want — I just want the ‘A.1’ part ...*

As is so often the case, Donald Arseneau came up with an elegant solution [1] shown below. But first, to set the context:

The L<sup>A</sup>T<sub>E</sub>X kernel `\@secCNTformat` macro typesets the number of a (sub-)section head, and takes one argument which is the name of the section head. Its default definition is:

```
\newcommand*{\@secCNTformat}[1]{%
  \csname the#1\endcsname\quad}
```

and in, for example, a `\subsubsection` it would be called as:

```
... \@secCNTformat{subsubsection}...
```

resulting in the code:

```
... \thesubsubsection\quad...
```

Donald proposed:

```
\makeatletter
\renewcommand*{\@secCNTformat}[1]{%
  \@ifundefined{#1prefix}{}%
    {\csname #1prefix\endcsname\ }%
  \csname the#1\endcsname. \quad}
\makeatother
\renewcommand*{\thesubsection}{%
  \Alph{section}.\arabic{subsection}}
\newcommand*{\subsectionprefix}{Activity}
```

As well as putting ‘Activity’ before subsection head numbers it also has the effect of putting a ‘.’ at the end of every sectional number. Using the above will result in `\section` heads like ‘2. Title’, `\subsection` heads like ‘Activity A.2. Title’ and `\subsubsection` heads like ‘A.2.3. Title’.

If the ‘.’ after every sectional number is not required this can be dealt with by extending Donald’s code to cater for putting something specific after the heading number, which can then be different for each section level:

```
\makeatletter
\renewcommand*{\@secCNTformat}[1]{%
  \@ifundefined{#1prefix}{}%
    {\csname #1prefix\endcsname\ }%
  \csname the#1\endcsname
  \@ifundefined{#1postfix}{}%
    {\csname #1postfix\endcsname}\quad}
\makeatother
\renewcommand*{\thesubsection}{%
  \Alph{section}.\arabic{subsection}}
\newcommand*{\subsectionprefix}{Activity}
\newcommand*{\subsectionpostfix}{.}
```

Using the above will result in `\section` heads like ‘2 Title’, `\subsection` heads like ‘Activity A.2. Title’ and `\subsubsection` heads like ‘A.2.3 Title’.

## References

- [1] Donald Arseneau. Re: Not using all of the reference. Post to `comp.text.tex` newsgroup, 1 November 2009.
- [2] Ulrich Diez. Re: a new `\verba` command. Post to `comp.text.tex` newsgroup, 23 October 2009.
- [3] Paul Isambert. Re: automatic line break within verbatim-environment. Post to `comp.text.tex` newsgroup, 10 April 2009.
- [4] David Kastrup. The preview package for L<sup>A</sup>T<sub>E</sub>X, 2010. Available on CTAN in `latex/macros/contrib/preview`.
- [5] Lars Madsen. Re: Recall a list item and print it later in the document. Post to `comp.text.tex` newsgroup, 14 September 2009.
- [6] Sebastian Rahtz and Heiko Oberdiek. Hypertext marks in L<sup>A</sup>T<sub>E</sub>X: a manual for hyperref, 2010. Available on CTAN in `latex/macros/contrib/hyperref`.
- [7] Will Robertson. Re: PDFs with a (very) variable page length. Post to `comp.text.tex` newsgroup, 12 October 2009.
- [8] Peter Wilson. Glisterings: Repetition, rectangular text. *TUGboat*, 30(2):287–289, 2009.

◇ Peter Wilson  
 20 Sovereign Close  
 Kenilworth, CV8 1SQ  
 UK  
 herries dot press (at)  
 earthlink dot net