Variable width boxes in \LaTeX

Simon Law

Seasoned \LaTeX\ users are familiar with the default box commands: \texttt{\makebox}, \texttt{\framebox} and \texttt{\parbox}. They are the building blocks for page layout, and are commonly used. After all, being able to create boxes allows a typesetter great flexibility in positioning objects on a page. Figure 1 illustrates a simple use of \texttt{\parbox}.

Hello brave world
Goodbye cruel world

\textbf{Figure 1:} Using \texttt{\parbox} to position text

\section{Traditional \texttt{\parbox}}

As you can see, I was able to align the two boxes so that each would be aligned. Looking at the source code in Figure 2, you’ll see that I had to manually specify the box widths.

\begin{verbatim}
\parbox[t]{1cm}{Hello\brave\world} \\
\parbox[b]{1.5cm}{Goodbye\cruel\world}
\end{verbatim}

\textbf{Figure 2:} \texttt{\parbox} source code

Of course, guessing the width of the longest line gets tedious. You can try using \texttt{\settowidth} on the longest line, but that might change as your text changes.

\section{Using \texttt{\pbox}}

In order to automatically determine the width of the box, we will use the \texttt{\pbox}\textsuperscript{1} package. It provides the \texttt{\pbox} command, which is analogous to the \texttt{\mbox} command. In Figure 3, I typeset the same text using \texttt{\pbox} instead.

\begin{verbatim}
\pbox[t]{\textwidth}{Hello\brave\world} \\
\pbox[b]{\textwidth}{Goodbye\cruel\world}
\end{verbatim}

\textbf{Figure 3:} \texttt{\pbox} source code

The syntax for \texttt{\pbox} is quite similar to that of \texttt{\parbox}. You must provide the maximum width of the box \textit{(max-width)} and the contents \textit{(text)}:

\begin{verbatim}
\pbox[\texttt{pos}][\texttt{height}][\texttt{inner-pos}]{\texttt{max-width}}{\texttt{text}}
\end{verbatim}

By default, the centre of each box will be vertically aligned. However, the three optional arguments allow you to align the \texttt{\pbox} as necessary. These options work exactly like their \texttt{\parbox} counterparts.

\section{Now with \texttt{\minipage}}

This works well for simple paragraphs, where environments need not be embedded. However, once you start needing the features of the \texttt{\minipage} environment, you begin to run into the same problems. David Arseneau has solved this problem with his \texttt{\varwidth}\textsuperscript{2} package.

An example use would be to centre a \texttt{\verbatim} environment. This is normally done in a \texttt{\minipage} because the \texttt{\verbatim} environment left-flushes all its text against the left margin. In order to use the \texttt{\minipage}, you still have to figure out the width of its contents and specify it manually.

\begin{verbatim}
#include <stdio.h>
int main()
{
  printf ("Hello world!\n");
  return 0;
}
\end{verbatim}

\textbf{Figure 4:} Centered source code example

Figure 4 shows a snippet of source code that is representative of a sample in an article or a textbook. The code in Figure 5 illustrates how to typeset this without manually determining the width.

\begin{verbatim}
\centering
\begin{varwidth}{\textwidth}
\begin{verbatim}
#include <stdio.h>
int main()
{
  printf ("Hello world!\n");
  return 0;
}
\end{verbatim}
\end{varwidth}
\end{verbatim}

\textbf{Figure 5:} \texttt{\varwidth} source code

\section{Conclusion}

Both the \texttt{\pbox} and \texttt{\varwidth} packages are useful extensions to standard \LaTeX\\textsuperscript{2e}. They allow typesetters to place boxes and minipages throughout their documents without the need for guessing widths.

\hspace{1cm} Simon Law
\hspace{1cm} sfllaw@law.yi.org
\hspace{1cm} \url{http://www.law.yi.org/~sfllaw/}

\begin{footnotesize}
\textsuperscript{1} \url{http://www.ctan.org/tex-archive/macros/latex/contrib/pbox/}
\textsuperscript{2} \url{http://www.ctan.org/tex-archive/macros/latex/contrib/misc/varwidth.sty}
\end{footnotesize}