Dashed and layered boxes

Reuben Thomas
rrt@sc3d.org
11th December 2001

Abstract
dashbox provides new commands similar to \framebox and \fbox to typeset dashed and layered boxes.

1 User interface

The following commands are provided:

\dbox{text} works like \fbox, but the box is drawn with ![dashed lines].

\dashbox[width][pos]{text} works like \framebox, but the box is drawn with ![dashed lines].

\lbox[layers]{text} draws a ![stack of boxes] around its contents, with the number of layers given by the first parameter (default 2).

\dlbox[layers]{text} works like \lbox, but the boxes are drawn with ![dashed lines].

The following style parameters are available:

\dashlength \dashlength gives the length of a dash plus the following gap. The default is 6pt.

\dashdash \dashdash gives the length of a dash. The default is 3pt.

\layersize \layersize gives the protrusion of each layer below the previous one. The default is \dashdash.

The following standard parameters are also observed:

\fboxrule \fboxrule gives the width of the dashes.
\fboxsep \fboxsep gives the separation between the box and text inside it.
2 Implementation

2.1 Preliminaries
Make sure we’ve got what we need, and announce the package.

\begin{verbatim}
⟨∗package⟩
\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{dashbox}[2001/12/11 v1.14 Dashed and layered boxes]
\RequirePackage{calc}
\RequirePackage{ifthen}
\end{verbatim}

2.2 Style parameters
Define and give the default values of the style parameters.

\dashlength
\begin{verbatim}
\newlength{\dashlength} \setlength{\dashlength}{6pt}
\end{verbatim}
\dashdash
\begin{verbatim}
\newlength{\dashdash} \setlength{\dashdash}{3pt}
\end{verbatim}
\layersize
\begin{verbatim}
\newlength{\layersize} \setlength{\layersize}{\dashdash}
\end{verbatim}

2.3 Dashes
We need two new commands for drawing horizontal and vertical dashes.

\hd@shrule \hd@shrule takes one argument, the rule’s width. The thickness of the dash is given by \fboxrule.
\begin{verbatim}
\newcommand{\hd@shrule}[1]{%  
\hbox to #1{\vrule height \fboxrule width \dashdash%  \cleaders\hbox to \dashlength{%  \vfill\rule{\fboxrule}{\dashdash}\vfill}  \ifthenelse{\lengthtest{#1 > 2\dashdash}}{\vrule height \fboxrule width \dashdash}{}%  }}
\end{verbatim}

\vd@shrule \vd@shrule takes one argument, the rule’s height. The thickness of the dash is given by \fboxrule.
\begin{verbatim}
\newcommand{\vd@shrule}[1]{%  \vbox to #1{\hrule height \dashdash width \fboxrule%  \cleaders\vbox to \dashlength{\hrule{\fboxrule}\vfill}\hrule%  \ifthenelse{\lengthtest{#1 > 2\dashdash}}{\hrule height \fboxrule width \dashdash}{}%  }}
\end{verbatim}
2.4 Dashed boxes

A private save box and some lengths are defined. \dashedsavebox is a box to
hold the contents of a dashed box. \dashedboxwidth is the box’s width, and
\dashedboxtotalheight is the height plus the depth.

\dashedsavebox
26 \newsavebox{\dashedsavebox}
\dashedboxwidth
27 \newlength{\dashedboxwidth}
\dashedboxtotalheight
28 \newlength{\dashedboxtotalheight}
\dashedbox
\makedashbox is where the work is actually done. It puts the box together piece
by piece. It requires \dashedboxwidth to be set by its caller.
29 \newcommand{\makedashbox}{%

The height plus depth of the box is calculated.
30 \setlength{\dashedboxtotalheight}{% 
  \dp{\dashedsavebox}+\ht{\dashedsavebox}+2\fboxsep+2\fboxrule}%

The box is raised an appropriate amount, and drawn in a b-aligned parbox.
32 \raisebox{-\fboxrule-\fboxsep-\dp{\dashedsavebox}}{%
33 \parbox[b]{\dashedboxwidth}{%

Inter-line and inter-paragraph skip are disabled.
34 \offinterlineskip%
35 \parskip=0pt%

The top line is drawn; the kern makes the left and right sides line up properly.
36 \hrule{\dashedboxwidth}%
37 \kern-\fboxrule%
38 \par%

The left-hand side is now drawn, in a parbox of the correct width.
39 \parbox{\fboxrule}{\vdashrule{\dashedboxtotalheight}}%

Now the inside of the box is set in a parbox, with \fboxsep space top and
bottom. The kerns add \fboxsep space at the left and right.
40 \kern\fboxsep%
41 \parbox{\wd{\dashedsavebox}}{%
  \vspace{\fboxsep}\usebox{\dashedsavebox}\vspace{\fboxsep}%
42 \kern\fboxsep%
The right-hand side is drawn just like the left-hand side, and the bottom just like the top.

\parbox{\textwidth}{\boxwidth{\boxrule}{\boxtop{\boxwidth{\boxrule}{\boxtop{\textwidth}}}}%}
\par
\kern-\fboxrule%
\hrulefill{\boxwidth}
\}
\hbox
\hbox is just a wrapper around \texttt{\m@kedashbox} which saves its argument and then calculates the width according to that of its argument.

\newcommand{\hbox}{%  
\sbox{\savebox}{#1}  
\setlength{\dashwidth}{\wd\savebox+2\fboxsep+2\fboxrule}  
\m@kedashbox}
\dashbox
\dashbox is partly taken from that for \texttt{\framebox}. Depending on whether any optional arguments are given, it either simply calls \hbox, or sets the width to that given and does the typesetting via savebox and \texttt{\m@kedashbox}.

\def{\dashbox}{\ifnum\the\numexpr\#1\relax>0\relax  
\savebox{\savebox}{\#1-2\fboxsep-2\fboxrule}  
\m@kedashbox}
\dashbox

2.5 Layers
Another series of private variables are required for layers: \texttt{\layersavebox} holds the text to be set in a layer, \texttt{\layerwidth} holds the total width of the layer, \texttt{\layerboxwidth} the width of the layer box, \texttt{\layertotalheight} the height plus depth of the layer, \texttt{\layerlineheight} the lift of the top right-hand line, and \texttt{\layeroffset} the lift of the layer below the baseline.

\layersavebox
\newsavebox{\layersavebox}

\layerwidth
\newlength{\layerwidth}

\layerboxwidth
\newlength{\layerboxwidth}

\layertotalheight
\newlength{\layertotalheight}

\layerlineheight
\newlength{\layerlineheight}
\texttt{\textbackslash m@kelayer} makes a solid layer.
\begin{verbatim}
\newcommand{\m@kelayer}[1]{%
The various lengths are calculated. The argument gives the number of the
layer, i.e. how far down it should be offset from its contents as a multiple of
\texttt{\textbackslash layersize}.
\setlength{\l@yertotalheight}{}
\setlength{\l@ayerlineheight}{}
\setlength{\l@yervoffset}{}
\setlength{\l@ayerboxwidth}{}
\parbox{\l@ayerwidth}{%
Inter-line and inter-paragraph spacing are turned off.
\offinterlineskip%
\parskip=0pt%
The contents of the layer is set first.
\usebox{\l@yersavebox}%
The extra “corner” is added on to the bottom right.
\rule{\l@ayerlineheight}{\textbackslash layersize}{\textbackslash fboxrule}%
\kern-\textbackslash fboxrule%
\rule{\l@yervoffset}{\textbackslash fboxrule}{\l@yertotalheight}%
\kern-\wd{\l@yersavebox}\kern-\textbackslash layersize\kern\textbackslash kern1\kern\textbackslash layersize
\rule{\l@yervoffset}{\textbackslash fboxrule}{\textbackslash layersize}%
\kern-\textbackslash fboxrule
\kern\textbackslash kern1
\kern-\textbackslash fboxrule
\rule{\l@yervoffset}{\l@ayerboxwidth}{\textbackslash fboxrule}%
}}
\end{verbatim}
\texttt{\textbackslash \textbackslash l@yer} draws a layer. The first argument gives the number of the layer, and the
second its contents.
\begin{verbatim}
\newcommand{\l@yer}[2]{%
\sbox{\l@yersavebox}{#2}%
\setlength{\l@yerwidth}{\wd{\l@yersavebox}+\textbackslash layersize}%
\m@kelayer{#1}}
\end{verbatim}
\texttt{\textbackslash m@kedashlayer} makes a dashed layer. The code is the same as for \texttt{\textbackslash m@kelayer}
except for the dash commands.
2.6 Stacks

Finally, the commands for drawing a stack of layers.

\l@yercount counts the number of layers drawn by the stack drawing command.

\l@yers draws a stack of layers; it is parametrized on the command used to draw a layer (third argument). The first argument is number of layers, and the second is the text to set. The layers are drawn in in a loop, using \l@yersavebox as an accumulator, and the result is typeset.
\begin{verbatim}
whiledo{not{\value{\yercount} > #1}}
   \sbox{\yersavebox}{#3{\value{\yercount}}{\usebox{\yersavebox}}}\stepcounter{\yercount}
\usebox{\yersavebox}
\end{verbatim}

\lbox and \dlbox are just wrappers for \yers. They both default to drawing two layers.

\begin{verbatim}
\newcommand{\lbox}[2][2]{\yers{#1}{#2}{\yer}}
\newcommand{\dlbox}[2][2]{\yers{#1}{#2}{\dlayer}}
\end{verbatim}