The \texttt{pspicture} package\textsuperscript{*}

D. P. Carlisle

16 June 1992

1 Introduction

\texttt{pspicture} is a re-implementation, and extension of, \LaTeX{}'s \texttt{picture} environment, using PostScript \texttt{special}'s. This has several advantages, mainly that lines of arbitrary slope and thickness may be specified, and there is no limit on the size of the circles that may be drawn\textsuperscript{1}.

One disadvantage is that the picture can no longer be previewed on a \texttt{dvi} previewer, such as \texttt{xdvi}. To help with this problem, a companion style option, \texttt{texpicture}, may be used while developing a document, this uses the standard picture commands as much as possible, and silently omits any picture objects that can not be drawn with standard \LaTeX{}.

A second disadvantage, is that a \texttt{dvi} file produced with \texttt{pspicture} will contain embedded \texttt{special} commands. These commands will only work with the driver program for which they were intended. This makes the \texttt{dvi} file less portable. \texttt{pspicture} will by default use \texttt{special}'s set up for Rokicki's \texttt{dvips} program, although it should be easy to modify the code to work with other PostScript drivers. A \texttt{DocStrip} option for a version of \texttt{dvi2ps} is included with this distribution.

1.1 Commands Available

\begin{description}
\item[\texttt{\circle}] Use as described in the \LaTeX{} book but with no maximum diameter. The thickness of the circle is altered by the \texttt{\linethickness} command. The size of the circle produced by \texttt{\circle} is not affected by \texttt{\linethickness}, so it is not the same as ‘filling in’ the circle drawn by \texttt{\circle}.

\item[\texttt{\circle*}] Use as described in the \LaTeX{} book, but as there is no maximum diameter for the circular arcs, the oval (in the absence of the optional [tr] etc) always consists of two semi-circular arcs joined by a pair of parallel lines. To obtain a ‘rectangle with rounded corners’ the oval command has a second optional argument (given first!).

\begin{verbatim}
\oval[20](100,200)[t]
\end{verbatim}

Produces the top half of an oval with quarter circles of radius 20*unitlength. If

\textsuperscript{*}This file has version number v2.02, last revised 1999/04/11/.

\textsuperscript{1}There is a certain amount of overlap between this style option and the widely available \texttt{eepic} option. However when I wrote the first version of this, in 1989, I was not aware of \texttt{eepic} and \texttt{pspicture} has been reasonably popular in Manchester, even though \texttt{epic} and \texttt{eepic} have been installed.
unitlength = 1pt then this is equivalent to the standard oval command. In general \\
\oval[R](x,y) uses circular arcs of radius \text{min}(R, x/2, y/2).

\line Use as described in the \TeX\ book but with no restriction on the available slopes. \\
The thickness of a sloping line is altered by the \linethickness command.

\vector New forms of the line and vector commands.
\put(x1,y1){\Line(x2,y2)} produces a line from (x1,y1) to (x1+x2,y1+y2) and similarly for \Vector.

\Curve Like \Line except that it produce a curve! \\
\put(x1,y1){\Curve(x2,y2){m}} produces a curve from (x1,y1) to (x1+x2,y1+y2). the amount of curvature is controlled by m but try 1 or −1 first. m does not have to be an integer. Negative numbers curve the opposite way to positive numbers.

\thinlines These commands alter the thickness of all lines including slanted lines and circular arcs.
\thicklines
\linethickness

\arrowlength A new command which specifies the size of the arrowhead drawn by the \vector and \Vector commands. Like \linethickness it does not get multiplied by \unitlength. At present the arrowhead is triangular. If a head with curved sides more like the standard \TeX head is required the definition of !A in pspicture.ps should be altered.

Other picture mode commands are not altered by this style, and so may be used, just as described in the \TeX book. These include: \put, \multiput, \makebox, \framebox, dashbox and \shortstack.
2 Examples

A picture built with \LaTeX's line and circle fonts.

The same picture built with PostScript \special's.

Some extra features not available using the standard picture mode.

3 pspicture.sty

First we set up the code that is specific to the driver program that is being used. If the driver can incorporate a header file, define \PS@header appropriately; \PS@special should expand to the format for inline PostScript code. The driver
should protect this code with a (g)save (g)restore pair. dvips is treated specially so that it will be the default driver if this file is used without being stripped. If you find definitions of these macros which work for the driver you use, email me, and I will include them in the next release.

\PS@header Code to include the PostScript header.
\PS@header{pspicture.ps}

\PS@special The format of the \special command for inline PostScript.
\PS@special{pspicture.ps}

\striptt Strip the final ‘pt’ off the string returned by \the.
\striptt{pspicture.ps}

\@circle Internal name for \circle.
\@circle{pspicture.ps}

\@dot Internal name for \circle*.
\@dot{pspicture.ps}

\line Line with a \LaTeX{} style slope specification.
\line{pspicture.ps}

\vector Line and arrow head with a \LaTeX{} style slope specification.
\vector{pspicture.ps}
\oval If no optional argument appears, use a default of maximum radius of \TeX’s maximum length.

\def\oval{% \\
@ifnextchar[{{
\@ov@l}{
count@=\maxdimen \divide\count@ by \unitlength \@ov@l\[
count@
}]}%}

\@ov@l Look for an optional tlbr argument.

\def\@ov@l[#1](#2,#3){% \\
@ifnextchar[{{\@oval[#1](#2,#3)}{\@oval[#1](#2,#3)\[
}}%}

\@oval The PostScript version of the \oval command will print each quarter of the oval separately, each quarter will only be printed if the appropriate argument is 1. An optional argument of t causes the arguments for the two bottom quarters to be set to 0, similarly, r causes the two left quarters to be set to 0. Thus an argument [tr] will set the bottom and left quarters to 0, resulting in only the top right quarter being printed.

\def\@oval[#1](#2,#3)[#4]{% \begingroup \\
\@tempdimb #1\unitlength \\
\@ovxx #2\unitlength \\
\@ovyy #3\unitlength \\
\def\r{\def\TL{0}\def\BL{0}} \\
\def\l{\def\TR{0}\def\BR{0}} \\
\def\t{\def\BL{0}\def\BR{0}} \\
\def\b{\def\TL{0}\def\TR{0}} \\
\@tfor\@tempa :=#4\do{\csname\@tempa\endcsname}% \\
\PS@special{%% \\
\BR\BL\TR\TL \\
\strippt\@wholewidth \\
\strippt\@tempdimb \\
\strippt\@ovxx \\
\strippt\@ovyy \\
!0}% \\
\endgroup}

\Line New style \Line command.

\def\Line(#1,#2){% \\
@ovxx #1\unitlength \\
@ovyy #2\unitlength \\
\PS@special{%% \\
\strippt\@wholewidth \\
\strippt\@tempdimb \\
\strippt\@ovxx \\
\strippt\@ovyy \\
!0}%

\Curve Not particularly good, but it will do for now.

\def\Curve(#1,#2)#3{% \\
@ovxx #1\unitlength \\
@ovyy #2\unitlength
\@ovyy #2\unitlength
\PS@special{%
  \strippt\wholewidth
  \strippt\@ovxx
  \strippt\@ovyy
  #3
  !C2}}
\Vector New style \Vector command.
\def\Vector(#1,#2){%
  \@ovxx #1\unitlength
  \@ovyy #2\unitlength
  \PS@special{%
    \strippt\@arrowlength
    \strippt\@wholewidth
    \strippt\@ovxx
    \strippt\@ovyy
    !V2}}
\@arrowlength Length of an arrow head.
\newdimen\@arrowlength
\arrowlength Set the length of an arrow head.
\def\arrowlength#1{\@arrowlength #1}
\arrowlength{8pt}
If this file is used as a .sty file without being stripped, we want to stop here. The \endinput must not be at the beginning of the line, or DocStrip will stop here as well!.
\endinput
(/package)

4 texpicture.sty

A dummy style file so that documents using pspicture.sty can be previewed or printed (as much as possible) using a dvi (not PostScript) previewer or printer driver.

Just change ‘pspicture’ to ‘texpicture’ in the \documentstyle options list.
\@warning{texpicture.sty in operation:^^J}@spaces
\LaTeX document with pspicture.sty before printing)
\Line Define all these new commands to silently gobble their arguments.
\Vector \def\Line(#1,#2){}
\arrowlength
\def\Vector(#1,#2){}
\arrowlength
\Curve \def\arrowlength#1{\arrowlength{#1}}
\def\Curve(#1,#2)#3{}
\@badlinearg If a vector or line is called with a slope specification that is not allowed by standard \LaTeX, \@badlinearg is called to produce the error message. We do not want to see these errors, so:
\def\@badlinearg{}
\oval Give the standard \oval command another optional argument (which will be ignored), to match the extra argument defined in pspicture.sty.

\oval
\oval
\oval
\oval
104 \let\oval\oval
105 \def\oval[#1]{\oval}
106 \def\oval{\ifnextchar[\oval{\oval}}
107 (/texty)

5 pspicture.ps

The PostScript header file for use with pspicture.sty. Probably this should use the PostScript dictionary mechanism, to keep identifiers local to this package, but for now, just give them names beginning with !.

!BP PostScript uses \TeX’s bp, that is 1/72 of an inch, not \TeX’s pt, 1/72.27 of an inch, but it is inconvenient to get \TeX to output in bp, so we need to scale the PostScript.
109 /!BP{
110 72 72.27 div dup scale
111 }def

!A Arrow head:

\arrow length !A

112 /!A{
113 newpath
114 0 0 moveto
115 dup neg dup .4 mul rlineto
116 .8 mul 0 exch rlineto
117 closepath
118 fill
119 }def

!V \vector(⟨x⟩,⟨y⟩)
\arrow length \line width ⟨x⟩ ⟨y⟩ ⟨len*unitlength⟩ !V

120 /!V{
121 /!BP
122 /!X exch def
123 /!y exch def
124 /!x exch def
125 newpath
126 0 0 moveto
127 !x 0 eq {0 !y 0 lt {!X neg}{!X} ifelse}
128 {!x 0 lt {!X neg}(!X)ife else % @wholewidth
129 lineto
130 setlinewidth % @wholewidth
131 currentpoint
132 stroke
133 translate
134 !y !x atan
135 rotate
136 !A % @arrowlength
137 }def
\line{(x),(y)}
\langle arrow length \rangle \langle line width \rangle \langle x \rangle \langle y \rangle \langle len*unitlength \rangle !L
138 \!/L{
139 !BP
140 /!X exch def
141 /!y exch def
142 /!x exch def
143 newpath
144 0 0 moveto
145 !x 0 eq {0 !y 0 lt={!X neg} {!X} ifelse}
146 {!x 0 lt={!X neg} {!X} ifelse !X !y mul !x abs div} ifelse
147 lineto
148 setlinewidth % @wholewidth
149 stroke
150 }def

!C \circle{\langle diam \rangle}
\langle line width \rangle \langle diam*unitlength \rangle !C
151 !/C{
152 !BP
153 0 0 3 2 roll
154 2 div 0 360 arc
155 setlinewidth % @wholewidth
156 stroke
157 }def

!D \circle*{\langle diam \rangle}
\langle diam*unitlength \rangle !D
158 !/D{
159 !BP
160 0 0 3 2 roll
161 2 div 0 360 arc fill
162 }def

!O \oval{\langle max-radius \rangle, \langle x \rangle, \langle y \rangle, \langle tlbr \rangle}
\langle br \rangle \langle bl \rangle \langle tr \rangle \langle tl \rangle
\langle line width \rangle \langle max-radius*unitlength \rangle \langle x*unitlength \rangle \langle y*unitlength \rangle !O
The first four arguments should be either 0, denoting that that quarter should not
be drawn, or 1, to draw a quarter oval.
163 !/O{
164 !BP
165 /!y exch 2 div def
166 /!x exch 2 div def
167 /!r exch !x !y
Ghostscript appears to have a min operator, so the following 2 lines could be coded
as min min, but it's not in the Ref. Manual, and it doesn't work on my printer!
168 2 copy gt {exch} if pop
169 2 copy gt {exch} if pop
170 def
171 setlinewidth % @wholewidth
172 1 eq
173 {newpath
174  !x neg 0 moveto
175  !x neg !y 0 !y !r arcto 4 {pop} repeat
176  0 !y lineto
177  stroke}if
178  1 eq
179  {newpath
180  !x 0 moveto
181  !x !y 0 !y !r arcto 4 {pop} repeat
182  0 !y lineto
183  stroke}if
184  1 eq
185  {newpath
186  !x neg 0 moveto
187  !x !y neg 0 !y neg !r arcto 4 {pop} repeat
188  0 !y neg lineto
189  stroke}if
190  1 eq
191  {newpath
192  !x 0 moveto
193  !x !y neg 0 !y neg !r arcto 4 {pop} repeat
194  0 !y neg lineto
195  stroke}if
196  }def

!V2 \Vector((x),(y))
  ⟨arrow length⟩ ⟨line width⟩ ⟨x*unitlength⟩ ⟨y*unitlength⟩ !V2
197  /!V2{
198  !BP
199  2 copy exch
200  atan
201  /a exch def
202  2 copy
203  newpath
204  0 0 moveto
205  lineto % <x*unitlength> <y*unitlength>
206  3 2 roll
207  setlinewidth % @wholewidth
208  stroke
209  translate % <x*unitlength> <y*unitlength>
210  a rotate
211  !A % @arrowlength
212  }def

!L2 \Line((x),(y))
  ⟨line width⟩ ⟨x*unitlength⟩ ⟨y*unitlength⟩ !L2
213  /!L2{
214  !BP
215  newpath
216  0 0 moveto
217  lineto % <x*unitlength> <y*unitlength>
218  setlinewidth % @wholewidth
219  stroke
220  }def
\Curve((x),(y))\{\pm\}
\langle \text{line width} \rangle \langle x*\text{unitlength} \rangle \langle y*\text{unitlength} \rangle \langle \pm \rangle !C2

221 /!C2{
222 !BP
223 /!s exch def
224 /!/y exch def
225 /!/x exch def
226 newpath
227 0 0 moveto
228 0 0
229 !x 2 div !y 10 div !s mul add
230 !y 2 div !x 10 div !s mul sub
231 !x !y
232 curveto
233 setlinewidth \% @wholewidth
234 stroke
235 }def
236 }/ps)