

# BLUe's Graphs

— Simplicity, aha! —

**Kees van der Laan**

Hunzeweg 57,  
9893 PB Garnwerd, The Netherlands  
cgl@rc.service.rug.nl

## Abstract

Graphics facilities for plain  $\text{\TeX}$  have been collected from *The  $\text{\TeX}$ book*, *manmac* and *gkpmac*, as part of *blue.fmt*. The *gkpmac* picture macros are handy and provide a subset of  $\text{\LaTeX}$ 's functionality. A database of pictures is proposed.

No interaction with *METAfont* (or *METApost*) nor the incorporation of encapsulated PostScript have been treated.

**Keywords:** Circles, curves, database, disks, education, graphics, format, lines, macro writing, pictures, plain  $\text{\TeX}$ , reusable software parts, selective loading, software engineering, splines, vectors.

## 1 Introduction

This paper is about graphics as part of BLUe's format. I'll discuss  $\text{\TeX}$ book's graphics, *manmac*'s drawing figures, and *gkpmac*'s picture environment with quadratic splines for arbitrary curves.<sup>1</sup>

Most of my diagrams, set earlier via  $\text{\LaTeX}$ 's picture environment, are included as test examples. BLUe started to think about a picture database similar to the database of references, et voilà.

### Why?

I made use of  $\text{\LaTeX}$  in the past mainly because of the picture environment facilities. Because of *gkpmac*'s subset of the  $\text{\LaTeX}$ 's picture environment in plain, I can get rid of  $\text{\LaTeX}$ 's general overhead. The audience in mind consists of those authors who

- practise (self-)publishing
- choose for English and ASCII
- adhere to  $\text{\TeX}$  formatting
- like stability, consistency, simplicity, portability, generality, flexibility, in short a lifetime tool
- favour an open, well-documented system
- prefer an extensible (formatting) language
- and support the PD software adage.

## Notations and definitions

*manmac.tex* stands for the macros used by Knuth for formatting his Computers and Typsetting series of books. *gkpmac.tex* stands for the macros used by Knuth and his co-authors to format *Concrete Mathematics*.

## 2 $\text{\TeX}$ book's graphics

In *The  $\text{\TeX}$ book* Appendix D Knuth states

' $\text{\TeX}$  is designed to put boxes together either horizontally or vertically, not diagonally. But that is not a serious limitation, because the use of negative spacing makes it possible to put things anywhere on the page.'

The following example from *The  $\text{\TeX}$ book* is IMHO the basis for  $\text{\LaTeX}$ 's picture environment.<sup>2</sup> A little further Knuth states

'If you enjoy fooling around making pictures instead of typesetting ordinary text,  $\text{\TeX}$  will be a source of endless frustration/amusement for you, because anything is possible, if you have suitable fonts.'

This was well-taken by Lamport when he created his line fonts to be used in  $\text{\LaTeX}$ 's picture environment. It is said that Knuth uses nowadays *METApost* next to Adobe's *photoshop*.  $\text{\TeX}$  is good at placing 'elements' within context. The other half of the twins—*METAfont*—is good at creating the graphical elements, don't forget that!

The examples with quarter circles, *The  $\text{\TeX}$ book* 389–390, the dragon figures, remind me of Papert's 'Turtle' graphics.

<sup>1</sup>No *METAfont* nor *METApost*, or the use of encapsulated PostScript.

<sup>2</sup>It is just a pity that the `—` as separator has been abandoned in  $\text{\LaTeX}$  and that parentheses and commas have been introduced as parameter separators. Of course it is tempting to adhere the coordinate notation from mathematics, but it assumes too much knowledge from the user about parameter separators.

### 3 Manmac's drawing figures

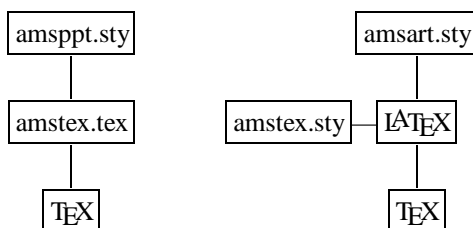
Basically macros for drawing hidden lines and a blank box are provided. The latter is handy in formatting a framed box.

```
\def\hidehrule#1#2{%Purpose:
%hrule height#1 depth#2
...}
\def\hidevrule#1#2{%Purpose:
%\vrule width#1+#2
...}
\def\makeblankbox#1#2{%Purpose:
%puts rules at the edge of a blank box
%whose dimensions are those of \box0
%(assuming nonnegative wd, ht, dp)
%#1 rule thickness outside
%#2 rule thickness inside
...}
%and special cases
\def\maketypebox{\makeblankbox{0pt}{1pt}}
\def\makelightbox{\makeblankbox{.2pt}{.2pt}}
```

Example (From Manmac BLUes)

```
$$\hbox{\vbox{%
\element{\fbox{amsppt.sty}}
\vconnector
\element{\fbox{amstex.tex}}
\vconnector
\element{\fbox{\TeX}}
}\qqquad\qqquad\qqquad\vbox{%
\element{\fbox{amsart.sty}}
\vconnector
\element{\llap{\fbox{amstex.sty}---}
\fbox{\LaTeX}}
\vconnector
\element{\fbox{\TeX}}
}}$$
```

I obtained



with the auxiliaries<sup>3</sup>

```
\def\strut{\vrule height2.5ex depth1ex
width0pt}
\def\fbox#1{\setbox0\hbox{\strut
$; $#1$, $}\leavevmode\rlap{\copy0}%
\makelightbox}%See appendix
\def\element#1{\hbox to15ex{\hss#1\hss}}
\def\vconnector{\element{\strut\vrule}} .
```

### 4 gkpmac's picture environment

It has been called in the file as 'Pictures (a subset of  $\text{\LaTeX}$ 's conventions).'  $\text{\LaTeX}$ 's line fonts have been used. Macros are provided for lines, vectors, curves, disks, circles, and

ovals. The lines and vectors have discrete orientations and can vary in thickness (2 states). The implementation is simpler than  $\text{\LaTeX}$ 's because of the omission of options, and comprises some 180 lines of code. Dimensionless units get their value via `\unitlength` to be initialized by the user. The documentation of the picture-environment in pseudo code comes with  $\text{\LaTeX}$ , and applies to this set too.

#### 4.1 As-is

To start it use `\gkpbeginpicture` and for closing use `\gkpendpicture`. The original definition reads<sup>4</sup>

```
\def\gkpbeginpicture(#1,#2)(#3,#4){%
%#1 x dimension
%#2 y dimension
%#3 x-offset
%#4 y-offset
...}
```

#### Line thickness

Default is `\thinlines` which can be toggled via the use of `\thicklines`. This does not apply to curves. Flexibility there is obtained via `\squineelem`,<sup>5</sup> the atomic element to be plotted repeatedly.

#### (multi)put

These are used to control the positioning of picture objects.

```
\def\put(#1,#2)#3{%
%#1 x-coordinate
%#2 y-coordinate
%#3 picture object
...}
\def\multiput(#1,#2)(#3,#4)#5#6{%
%#1 x-coordinate
%#2 y-coordinate
%#3 x-increase
%#4 y-increase
%#5 number of times
%#6 picture object
...}
```

#### Picture objects

Provided are: (h)box, line, vector, disk, circle, oval.

```
\def\makebox(#1,#2)#3{%
%#1 x-dimension
%#2 y-dimension
%#3 is put in hbox to#1\unitlength
% {\hss#3\hss}
...}
\def\line(#1,#2)#3{%abs(#1,#2)<=6
%#1 horizontal slope
%#2 vertical slope
%#3 horizontal size
...}
\def\vector(#1,#2)#3{%abs(#1,#2)<=4
%#1 horizontal slope
%#2 vertical slope
%#3 horizontal size
...}
```

<sup>3</sup>`\makeblankbox` is used in *The TeXbook* in exercise 11.5, for the skyline of words.

<sup>4</sup>Not conform  $\text{\LaTeX}$  because the offset parameters are obligatory. These two outer level names have been prefixed by `gkp`, to free the names `\beginpicture` and `\endpicture` while the original macros can still be accessed. The next best to a compatible extension.

<sup>5</sup>Introduced by me.

```
%
\def\disk#1{%
%#1 diameter <=.2inch
...}
%
\def\circle#1{%
%#1 diameter <=.5inch
...}
%
\def\oval(#1,#2){%
%#1 x-dimension
%#2 y-dimension
%centre set at reference point
...}
```

**Curves.** The method has been explained in the METAfont book p. 13. The functionality provided is similar to Lamport's Bezier.sty, although here a quadratic spline is used.

```
\def\squine(#1,#2,#3,#4,#5,#6){%
%#1, #2, #3 x-coordinates of consecutive points
%#4, #5, #6 y-coordinates of consecutive points
%To plot f(x) between x0 and x1
%\put(0, 0){\squine(x0,xm,x1,y0,ym,y1)}, with
%y0=f(x0), y1=f(x1)
%xm=(y0-y1+s1x1-s0x0)/(s1-s0)
%ym=(s0(s1x1-y1)-s1(s0x0-y0))/(s1-s0)
%s0=f'(x0), s1=f'(x1)
```

`\beginpic` `\endpic` can be used to obtain a vertically centered picture.<sup>6</sup>

Example (*Simplest use*)

```
\unitlengthlex
\gkpbeginpicture(2, 2)(0, 0)
\put(0,0){\markorigin}
\gkpendpicture
```

## 4.2 Mods

The `\gkpbeginpicture` macro has been adapted for

- the offsets as options with default the origin
- consistency with `blue.fmt`<sup>7</sup>
- allowing a database of pictures to be set up.

```
\newtoks\xdim \newtoks\ydim
\newtoks\xoffset \newtoks\yoffset
\newtoks\everypicture
\newtoks\thispicture
\xoffset{0}\yoffset{0}%defaults
%
\def\beginpicture{\bgroup
\the\everypicture\the\thispicture
\gkpbeginpicture(\the\xdim,\the\ydim)%
(\the\xoffset,\the\yoffset)}
%
\def\endpicture{\gkpendpicture\egroup
\global\thispicture{}}
```

Remark. I introduced an extra level of grouping to shield `\line`.

## Picturebase

To allow a picture database with dimensionless sizes included, provide as entries

```
\def<name>pic
{\bgroup%defaults
\unitlength=<dimen number>
\xdim{number}\ydim{number}
\beginpicture
...
\endpicture
\egroup}
```

## Selecting and loading

Imagine you have built up a number of pictures each consisting of a reasonable amount of picture instructions. Then you can either input the whole file and use only those entries needed (via `\loadallpictures`, because the common aspects have to be activated) or load selectively only those needed. I have used this selective loading before in 'BLUE's Bibliography,' and it can easily be used for this application of selective loading of pictures from a picturebase. The basic observation is that specified control sequences are redefined. This in contrast with the application of a format or style file where the file is not loaded when some names are already available, to prevent redefinition. My use is the opposite. I specify the names I like to be selected.

The use with `blue.fmt` is to supply the following in the preliminary part

```
\pictures{\<name-1>pic...<name-n>pic}
```

and to invoke the picture at the appropriate place in the script via

```
<name>pic
eventually preceded with \thispicture....
```

## 5 Examples

In the following examples I assume that the loading of the picture has already taken place in the preliminary part of the script. The origin has been printed via a + now and then.

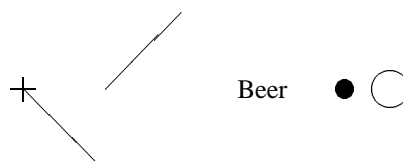
Example (*Some simple graphs.*)

```
\thispicture{\markorigintrue}
\simplepic
with result
```



Example (*Lines, text, circle and disk*)

```
\thispicture{\markorigintrue}
\linespic
with result
```



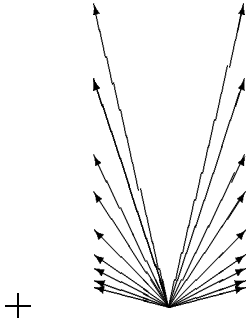
<sup>6</sup>The `\beginpic` name has been added to the original `\cpic`, for consistency reasons.

<sup>7</sup>To remind you: a two-part macro does not take arguments, but all is provided via `\everypicture`, respectively `\thispicture`.

Example (*Vectorbundle via gkp.mac*)

```
\thispicture{\markorigintrue}
\vectorbundlepic
```

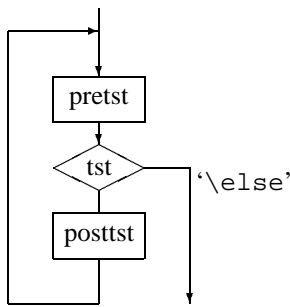
with result



Example (*Flow chart loop*)

```
\flowchartlooppic
```

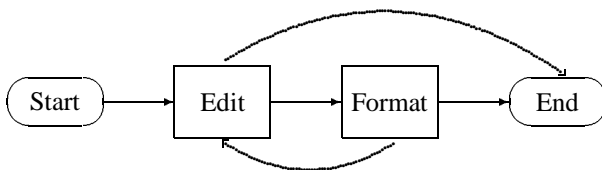
with result



Example (*Flow chart due to Furuta*)

```
\furutapic
```

with result



Note the use of PMA—poor man's arrowheads.

In the same spirit the following pictures have been realized.

- Salomon's Overview of  $\text{\TeX}$  etc.
- SGML- $\text{\TeX}$  relation
- NTG's Pie-chart

These have been displayed earlier in MAPS. The pictures could easily be transcribed with `\bezier` replaced by `\squine`.

## 6 Related work

$\text{\PCTeX}$  is definitely more complete. I have not studied that in detail of yet. It is not in the PD. For most of my needs at the moment `gkpic` is sufficient and small.

## 7 Acknowledgements

Włodek Bzyl made me aware of `gkpic`, although for a different purpose. Jos Winnink is kindly acknowledged for proofing the article and coercing it into MAPS style. Erik Frambach suggested to mark the origin in the examples. Thank you!

## 8 Conclusions

Manmac's `\makeblankbox` is handy.

For line diagrams `gkpic`'s picture macros are needed. Diagrams set earlier via  $\text{\LaTeX}$ 's picture environment could be adapted easily.

The placement of arrowheads—vectors of zero length—is tricky and can't be done easily.

A problem still is how to add these arrowheads to curves. In want for more general arrowheads PMA—poor man's arrowheads—have been introduced.

A database of pictures, in analogy with a bibliographical database, has been proposed.

## 9 References

The references are of course the  $\text{\TeX}$ book, the  $\text{\LaTeX}$  manual and the macro file `gkpic` used to typeset Concrete Mathematics.

The complete article is available—in `blue.fmt`—on request.

## 10 Appendix: `gkpic`'s graphics

```
% Pictures (a subset of \LaTeX's conventions)
1
2 \catcode'\@=11 % borrow the private macros
3 % of PLAIN (with care)
4 \newskip\hsssglue
5 \hsssglue=0pt plus 1fill minus 1fill
6 \def\hsss{\hskip\hsssglue}
7
8 \newdimen\unitlength \newdimen\linethickness
9 \newdimen\@picheight \newdimen\@xdim
10 \newdimen\@ydim \newdimen\@len \newdimen\@save
11 \newcount\@multicount \newcount\@xarg
12 \newcount\@yarg\newbox\@picbox \newbox\@mpbox
13
14 \font\tenln=line10 \font\tenlnw=line10
15 \font\tencirc=lcircle10
16 \font\tencircw=lcirclew10
17 \font\smallln=line10 scaled 483 % magstep-4
18
19 \def\thinlines{\let\linefont=\tenln
20 \let\circlefont=\tencirc
21 \linethickness=\fontdimen8\linefont}
22 \def\thicklines{\let\linefont=\tenlnw
23 \let\circlefont=\tencircw
24 \linethickness=\fontdimen8\linefont}
25 \thinlines
26
27 \def\gkpbeginpicture(#1,#2)(#3,#4){\begingroup
28 \@picheight=#2\unitlength \let\line=\@line
29 \setbox\@picbox=\hbox to#1\unitlength
30 \bgroup\kern-#3\unitlength
```

```

31 \lower#4\unitlength\hbox\bgroup
32 \ignorespaces}\let\picture\beginpicture
33 \def\gkpendpicture{\egroup\hss\egroup
34 \ht\@picbox=\@picheight \dp\@picbox=\z@
35 \leavevmode\box\@picbox\endgroup}
36
37 \def\put(#1,#2)#3{\raise#2\unitlength
38 \rlap{\kern#1\unitlength #3}\ignorespaces}
39
40 \def\multiput(#1,#2)(#3,#4)#5#6{\@multicount
41 =#5\@xdim=#1\unitlength \@ydim=#2\unitlength
42 \setbox\@mpbox=\hbox{#6}%
43 \loop\ifnum\@multicount>0
44 \raise\@ydim\rlap{\kern\@xdim
45 \unhcopy\@mpbox}%
46 \advance\@xdim#3\unitlength
47 \advance\@ydim#4\unitlength
48 \advance\@multicount\m@ne
49 \repeat\ignorespaces}
50
51 %makebox is replaced because its reference
52 %point does not coincide with LaTeX's
53 \def\makebox(#1,#2)#3{\setbox\@picbox=
54 % \hbox to#1\unitlength{\hss#3\hss}%
55 % \advance\@ydim-\@dp\@picbox
56 % \ht\@picbox=#2\unitlength
57 % \dp\@picbox=\z@
58 % \leavevmode\lower.5\@ydim\box\@picbox}
59 \def\makebox(#1,#2)#3{%
60 \hbox to#1\unitlength{\hss
61 \vbox to#2\unitlength{\vss
62 \hbox{#3}\vss}\hss}}
63 %LaTeX's framebox is added
64 \def\framebox(#1,#2)#3{\setbox0=
65 \hbox to#1\unitlength{\hss
66 \vbox to#2\unitlength{\vss
67 \hbox{#3}\vss}\hss}%
68 \makelightbox\kern-\wd0\box0}
69
70 \newif\ifneg
71 \def\@line(#1,#2)#3{\@xarg=#1 \@yarg=#2
72 \@len=#3\unitlength \leavevmode
73 \ifnum\@xarg<0 \reverseline \else
74 \negfalse \@ydim=\z@\fi
75 \ifnum\@xarg=0 \vline
76 \else\ifnum\@yarg=0 \hline
77 \else\@sline\fi\fi
78 \ifneg\kern-\@len\else\@save=\@ydim\fi}
79 \def\reverseline{\negtrue \kern-\@len
80 \xarg=-\@xarg
81 \@ydim=\@len \multiply\@ydim\@yarg
82 \divide\@ydim\@xarg \@yarg=-\@yarg}
83
84 \def\@hline{\vrule height.5\linethickness
85 depth.5\linethickness width\@len}
86 \def\@vline{\kern-.5\linethickness\vrule
87 width\linethickness
88 \ifnum\@yarg<0 height\z@ depth\else
89 depth\z@ height\fi\@len
90 \kern-.5\linethickness}
91
92 \def\@sline{\setbox\@picbox=\hbox{\linefont
93 \count@=\@xarg \multiply\count@ 8
94 \ifnum\@yarg>0 \advance\count@\@yarg
95 \advance\count@-9
96 \else \advance\count@-\@yarg
97 \advance\count@ 55 \fi
98 \immediate\write0{count@=\the\count@}%
99 \char\count@}%
100 \ifnum\@yarg<0 \@picheight=-\ht\@picbox
101 \advance\@ydim\@picheight
102 \else \@picheight=\ht\@picbox \fi
103 \@xdim=\wd\@picbox \@save=\@ydim
104 %\hbox to0pt{\hss$\bullet$\hss}%
105 \loop\ifdim\@xdim<\@len
106 \raise\@ydim\copy\@picbox
107 \advance\@xdim\wd\@picbox
108 \advance\@ydim\@picheight
109 \repeat
110 \advance\@xdim-\@len \kern-\@xdim
111 \multiply\@xdim\@yarg \divide\@xdim\@xarg
112 \advance\@ydim-\@xdim
113 \ifdim\@len<\ht\@picbox%Mod cgl June 94
114 \kern\wd\@picbox\else
115 \raise\@ydim\box\@picbox\fi}
116
117 \def\vector(#1,#2)#3{\@line(#1,#2){#3}%
118 \ifnum\@xarg=0 \vvector
119 \else\ifnum\@yarg=0 \hvector
120 \else\svector\fi\fi}
121 \def\@hvector{\ifneg\rlap{\linefont\char
122 27}\else\smash{\llap{\linefont\char45}}\fi
123 } % we have to smash because of font bug
124 \def\@vvector{\ifnum\@yarg<0
125 \raise-\@len\rlap{\linefont\char63}%
126 \else\setbox\@picbox=\rlap{\linefont
127 \char54}\advance\@len-\ht\@picbox
128 \raise\@len\box\@picbox\fi}
129
130 \def\@svector{\setbox\@picbox=\hbox
131 to\z@\linefont
132 \ifnum\@yarg<0 \count@=55 \@yarg=-\@yarg
133 \else\count@=-9 \fi
134 \ifneg\multiply\@xarg16 \multiply\@yarg2
135 \else\hss % \llap
136 \ifnum\@xarg>2 \multiply\@xarg9
137 \multiply\@yarg2
138 \advance\count@29
139 \else\ifnum\@yarg>2 \multiply\@xarg16
140 \multiply\@yarg9
141 \advance\count@-20
142 \else\multiply\@xarg24 \multiply\@yarg3
143 \fi\fi\fi
144 \advance\count@\@xarg
145 \advance\count@\@yarg \char\count@
146 \ifneg\hss\fi}% \rlap
147 \raise\@save\box\@picbox}
148
149 \def\disk#1{\@len=#1\unitlength
150 \count@='160 \@diskcirc}
151 \def\circle#1{\@len=#1\unitlength
152 \count@='140 \@diskcirc}
153 \def\@diskcirc{\setbox\@picbox=
154 \hbox{\circlefont\char\count@}%
155 \@xdim=\wd\@picbox \leavevmode
156 \ifdim\@len>15.499\@xdim \@bigdc
157 \else \@smalldc\fi}
158 \def\@bigdc{\ifnum\count@<'160 \@bigcirc
159 \else \@len=15\@xdim \@diskcirc\fi}
160 \def\@smalldc{\advance\@len-.5\@xdim
161 \loop\ifdim\@xdim<\@len
162 \advance\count@\@one
163 \advance\@xdim\wd\@picbox
164 \repeat
165 \hbox{\circlefont\char\count@}}
166 \def\@bigcirc{\circlefont\count@=15
167 \setbox\@picbox=\hbox{\char\count@
168 }\@xdim=\wd\@picbox
169 \ifdim\@len>2.5\@xdim \@len=2.5\@xdim\fi
170 \advance\@len-.125\wd\@picbox
171 \loop\ifdim\@xdim<\@len
172 \advance\count@4
173 \advance\@xdim.25\wd\@picbox
174 \repeat
175 \@ydim=.5\@xdim
176 \advance\@ydim.5\linethickness
177 \setbox\@picbox=\vbox{\hbox{\char
178 \count@\advance\count@-3\char\count@}%

```

```

179 \nointerlineskip
180 \hbox{\advance\count@m@ne\char
181 \count@advance\count@m@ne
182 \char\count@}}%
183 \kern-\@ydim\lower\@ydim\box\@picbox}}
184
185 \newif\ifovalt1 \newif\ifovaltr
186 \newif\ifovalbl \newif\ifovalbr
187 \ovalt1true \ovaltrtrue
188 \ovalbltrue \ovalbrtrue
189 \def\oval(#1,#2){\@xdim=#1\unitlength
190 \@ydim=#2\unitlength
191 {\circlefont
192 \setbox\@picbox=\hbox{\char0}
193 \ifdim\@xdim<\wd\@picbox
194 \@xdim=\wd\@picbox\fi
195 \ifdim\@ydim<\wd\@picbox
196 \@ydim=\wd\@picbox\fi
197 \@save=\@xdim
198 \ifdim\@ydim<\@save \@save=\@ydim \fi
199 \count@=39
200 \loop \setbox\@picbox=\hbox{\char
201 \count@}\ifdim\@save<\wd\@picbox
202 \advance\count@-4
203 \repeat
204 \setbox\strutbox=\hbox{\vrule height\ht
205 \@picbox depth\dp\@picbox width\z@
206 \kern\wd\@picbox}%
207 \@save=.5\wd\@picbox
208 \advance\@save-.5\linethickness
209 \setbox0=\hbox to\@xdim{\ifovalt1
210 \char\count@else\strut\fi
211 \kern-\@save\leaders\hrule height
212 \ifovalt1\linethicknesselse\z@\fi
213 \hfil
214 \leaders\hrule height\ifovaltr
215 \linethicknesselse\z@\fi
216 \hfil\kern\@save
217 \ifovaltr\advance\count@-3
218 \char\count@else\strut\fi
219 \kern-\wd\@picbox}%
220 \advance\count@m@ne
221 \setbox2=\hbox to\@xdim{\ifovalbl
222 \char\count@\else\strut\fi
223 \kern-\@save\leaders\hrule height
224 \ifovalbl\linethicknesselse\z@\fi
225 \hfil
226 \leaders\hrule height
227 \ifovalbr\linethicknesselse
228 \z@\fi\hfil\kern\@save
229 \ifovalbr\advance\count@m@ne
230 \char\count@\else\strut\fi
231 \kern-\wd\@picbox}%
232 \@save=\@ydim
233 \advance\@save-\wd\@picbox
234 \divide\@save 2
235 \setbox\@picbox=\vbox{\box0
236 \nointerlineskip
237 \hbox to\@xdim{\vrule height\@save
238 width\ifovalt1\linethickness
239 \else\z@\fi
240 \hfil\ifovaltr\vrule
241 width\linethickness
242 \kern-\linethickness\fi}%
243 \nointerlineskip
244 \hbox to\@xdim{\vrule height\@save
245 width\ifovalbl\linethickness
246 \else\z@\fi
247 \hfil\ifovalbr\vrule
248 width\linethickness
249 \kern-\linethickness\fi}%
250 \nointerlineskip\box2}%
251 \@save=.5\@ydim
252 \advance\@save.5\linethickness
253 \leavevmode
254 \kern-.5\@xdim
255 \kern-.5\linethickness
256 \lower\@save\box\@picbox}}
257
258 \def\beginpic#1\endpic{\vcenter{\hbox
259 {picture#1\endpicture}}}}
260
261 % Squines (quadratic splines)
262 % example of use: to plot f(x) between
263 % x0 and x1, you can say
264 % \put(0,0){\squine(x0,xm,x1,y0,ym,y1)},
265 % where y0=f(x0), y1=f(x1)
266 % xm=(y0-y1+s1x1-s0x0)/(s1-s0),
267 % ym=(s0(s1x1-y1)-s1(s0x0-y0))/(s1-s0),
268 % s0=f'(x0), and s1=f'(x1).
269
270 \newdimen\@xi \newdimen\@xii
271 \newdimen\@xiii \newdimen\@xiv
272 \newdimen\@xpt \newdimen\@xoldpt
273 \newdimen\@yi \newdimen\@yii
274 \newdimen\@yiii \newdimen\@yiv
275 \newdimen\@ypt \newdimen\@yoldpt
276 \def\squineelem{.}
277 %\def\squineelem{\tencirc q}
278 \def\squine(#1,#2,#3,#4,#5,#6){\setbox
279 \@picbox\hbox{\squineelem}%
280 \global\@xoldpt=#1\unitlength
281 \global\@yoldpt=#4\unitlength
282 \kern\@xoldpt
283 \@xi=\@xoldpt
284 \@xii=#2\unitlength
285 \@xiii=#3\unitlength
286 \@yi=\@yoldpt
287 \@yii=#5\unitlength
288 \@yiii=#6\unitlength
289 \squinerec
290 \@xpt=#3\unitlength
291 \@ypt=#6\unitlength
292 \@addpoint
293 \raise\@ypt\copy\@picbox}
294 \newif\iffar
295 \def\squinerec{\farfalse
296 \testnear\@xi\@xiii \testnear\@yi\@yiii
297 \iffar \decast \fi}
298 \def\testnear#1#2{\@save=#1\advance
299 \@save-#2%
300 \ifdim\@save<\z@ \@save=-\@save\fi
301 \ifdim\@save>\p@ \fartrue \fi}
302 \def\decast{\@xpt=\@xi
303 \advance\@xpt\@xii \divide\@xpt2
304 \advance\@xii\@xiii \divide\@xii2
305 \@xiv=\@xpt
306 \advance\@xiv\@xii \divide\@xiv2
307 \@ypt=\@yi
308 \advance\@ypt\@yii \divide\@ypt2
309 \advance\@yii\@yiii \divide\@yii2
310 \@yiv=\@ypt
311 \advance\@yiv\@yii \divide\@yiv2
312 \begingroup
313 \@xii=\@xpt \@xiii=\@xiv
314 \@yii=\@ypt \@yiii=\@yiv
315 \squinerec
316 \endgroup
317 \@xpt=\@xiv \@ypt=\@yiv \@addpoint
318 \@xi=\@xiv \@yi=\@yiv \squinerec
319 }
320 \def\@addpoint{%
321 %\message{(\the\@xpt,\the\@ypt)}%
322 \global\advance\@xoldpt-\@xpt
323 \wd\@picbox=-\@xoldpt
324 \raise\@yoldpt\copy\@picbox
325 \global\@xoldpt=\@xpt
326 \global\@yoldpt=\@ypt}

```

327	\catcode'\@=12	%	(and \line).....27-32
328	\endinput	%\gkpendpicture.....	33-35
329	\def\vectorhead(#1,#2){\@xarg=#1 \@yarg=#2	%\beginpic, \endpic.....	258-259
330	\setbox\@picbox=\hbox{\linefont	%\put.....	37-38
331	\count@=\@xarg \multiply\count@ 8	%\multiput.....	40-49
332	\ifnum\@yarg>0 \advance\count@\@yarg	%Picture elements	
333	\advance\count@-9	%\makebox.....	59-62
334	\else \advance\count@-\@yarg	%\framebox.....	64-68
335	\advance\count@ 55 \fi	%\@line.....	71-78
336	%\immediate\write0{count@=\the\count@}%	%\reverseline.....	79-82
337	\char\count@}}	%\hline.....	84-85
338	%Additions%\vrblin399	%\@vline.....	86-90
400	\newtoks\xdim \newtoks\ydim	%\@sline.....	92-115
410	\newtoks\xoffset \newtoks\yoffset	%\vector.....	117-120
402	\newtoks\everypicture	%\@hvector.....	121-123
403	\newtoks\thispicture	%\@vvector.....	124-128
404	\xoffset{0}\yoffset{0}%defaults	%\@svector.....	130-147
405	\let\gkpbeginpicture\beginpicture	%\disk.....	149-150
406	\def\beginpicture{\bgroup	%\circle.....	151-152
407	\the\everypicture\the\thispicture	%\diskcirc.....	153-157
408	\gkpbeginpicture(\the\xdim,\the\ydim)%	%\@bigdc.....	158-159
409	(\the\xoffset,\the\yoffset)	%\@smalldc.....	160-165
410	}	%\@bigcirc.....	166-183
411	\let\gkpendpicture\endpicture	%\oval.....	189-256
412	\def\endpicture{\gkpendpicture\egroup	%Curves	
413	\global\thispicture{}}	%\squineelem.....	276-277
414	%	%\squine.....	278-293
	%Contents Picture environm. gkpmac	%\squinerec.....	295-297
	%\unitlength.....	%\testnear.....	298-301
	%\linethickness.....	%\decast.....	302-320
	%\@picheight.....	%\@addpoint.....	321-328
	%\@xdim, \@ydim.....	%Additions, compatible blue.fmt conventions	
	%\@xarg, \@yarg, \@len.....	%Newtoks	
	%\@save.....	%\xdim, \ydim, xoffset, \yoffset..	400-401
	%\@picbox.....	%\everypicture, \thispicture.....	402-403
	%\@mpbox.....	%\beginpicture.....	406-410
	%\@xi, \@xii, \@xiii, \@xiv.....	%\endpicture.....	411-412
	%\@xpt, \@xoldpt.....	%	
	%\@yi, \@yii, \@yiii, \@yiv.....	%Modifications	
	%\@ypt, \@yoldpt.....	%Jun 94 \makebox adapted to LaTeX.	
	%Newifs	% \framebox added.	
	%\ifneg.....	% \@sline adapted for case	
	%\ifovaltl, \ifovaltr.....	% \@len<\ht\@picbox (suppress line	
	%\ifovalbl, \ifovalbr.....	% element but kern conditionally.)	
	%\iffar.....	% The 'dot' element in \squine has been	
	%Fonts	% parameterized and initialized	
	%\tenln, \tenlnw.....	% differently.	
	%\tencirc, \tencircw.....	% blue.fmt environment conventions added.	
	%\smallln.....		
	%Size switching		
	%\thinlines, \thinlines.....		
	%Picture environment		
	%\gkpbeginpicture		

## 11 Appendix: pic.dat

This file is available from the file servers.