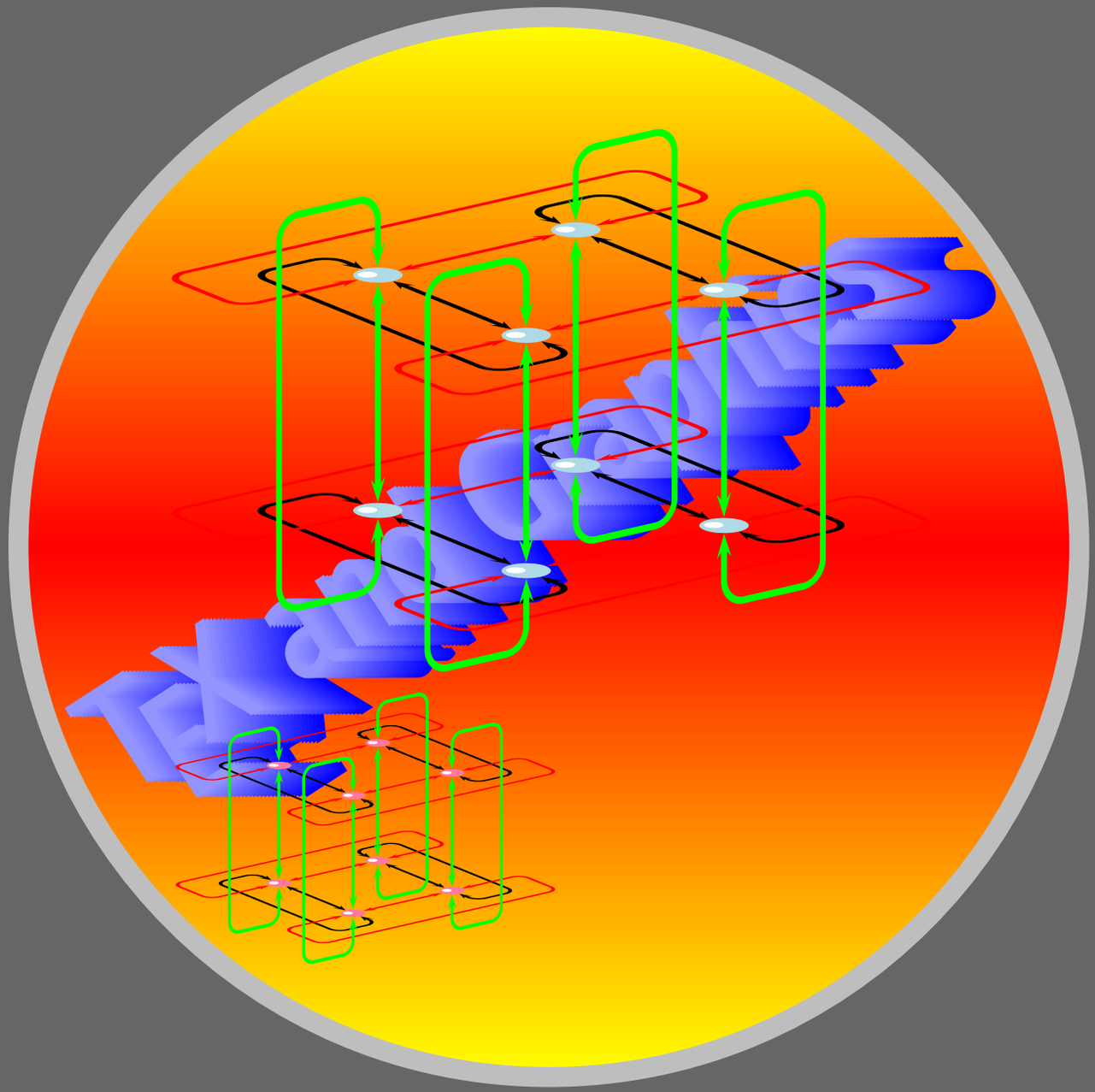


Custom Graphics

Boxing Text
Clipping
Rotating and scaling
Text along a path
Text as graphic

Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TeX Users Group
This document is generated by pdfTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

9. Tricks with Text

In our discussions so far, we've been focusing on *graphic objects* and we've treated *text* only incidentally in Chapter 6, as labels in pictures. We now see how text can be manipulated in various ways using PSTricks.

Boxing Text
Clipping
Rotating and scaling
Text along a path
Text as graphic

Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TeX Users Group
This document is generated by pdfTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

9.1. Boxing Text

LaTeX has various macros for putting text in boxes (or putting boxes around text) and PSTricks defines its own boxing macros. The advantage of using these is the ease of adorning these boxes using colors, shadows and so on. The simplest of such commands is the `\psframebox` as in the example below:

```
\psframebox[fillstyle=solid,%
            fillcolor=Cyan,%
            linecolor=RoyalBlue]%
            {\color{Red}
             \LARGE\bfseries
             Text In A Box}
```

Text In A Box

The distance between the sides of the box and the enclosed text is controlled by the `framesep` parameter. By default, its value is 3 point, but as with other parameters, can be set to any desired value, as shown in the next example:

```
\psframebox[framesep=10pt,%
            fillstyle=solid,%
            fillcolor=Cyan,%
            linecolor=RoyalBlue]%
            {\color{Red}
             \LARGE\bfseries
             Text In A Box}
```

Text In A Box

A variant of the `\psframebox` is the `\psdblframebox` which, as the name indicates, doubles each line of the frame

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TeX Users Group
This document is generated by pdfTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\psdblframebox[framesep=10pt,%
               fillstyle=solid,%
               fillcolor=Cyan,%
               linecolor=RoyalBlue,%
               doublecolor=Apricot,%
               doublesep=3pt]%
              {\color{Red}
               \LARGE\bfseries
               Text In A Box}
```

Text In A Box

Recall that the `doublesep` parameter determines the width of the space between the double lines and the `doublesep` the color of this space, as mentioned in Chapter 3. The default value of `doublesep` for the `\psdblframebox` is `\pslinewidth` and the default value of `doublecolor` is white.

Another variant is the `\psshadowbox` which, obviously enough, draws a (single) frame with a shadow, as shown below:

```
\psshadowbox[framesep=10pt,%
             fillstyle=solid,%
             fillcolor=Cyan,%
             linecolor=RoyalBlue,%
             shadowcolor=Blue,%
             shadowsize=5pt]%
            {\color{Red}
             \LARGE\bfseries
             Text In A Shadow Box}
```

Text In A Shadow Box

Note that the parameters `shadowsize` and `shadowcolor` are discussed in Chapter 3.

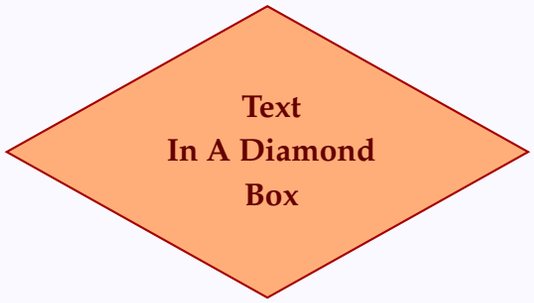
If you are tired of plain old rectangular boxes, you can try `\psdiabox`, which draws a diamond shaped box:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

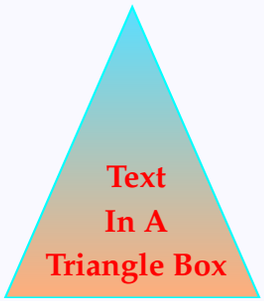
The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\psdiabox[fillstyle=solid,%
          fillcolor=Apricot,%
          linecolor=Mahogany]%
         {\color{Brown}
          \large\bfseries
          \renewcommand{%
            \arraystretch}{1.2}
          \begin{tabular}{c}
            Text\\
            In A Diamond\\
            Box
          \end{tabular}}
```



or `\pstribox`, which draws a triangular box:

```
\pstribox[fillstyle=gradient,%
          gradbegin=CornflowerBlue,%
          gradend=Apricot,%
          gradmidpoint=1,%
          linecolor=Cyan]%
         {\color{Red}
          \large\bfseries
          \renewcommand{%
            \arraystretch}{1.2}
          \begin{tabular}{%
              @{\hspace{-15pt}}c@{\hspace{-15pt}}}
                Text\\
                In A\\
                Triangle Box
          \end{tabular}}
```



Recall that the `gradient` style of filling requires the pst-grad package, as explained in Chapter 2.

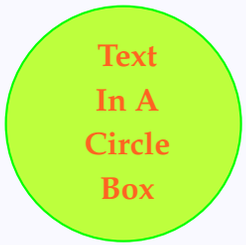
For those who are inclined towards curves than angles, there's a `\pscirclebox`:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\pscirclebox[fillstyle=solid,%
             fillcolor=SpringGreen,%
             linecolor=Green]%
            {\color{Orange}
             \large\bfseries
             \renewcommand{%
               \arraystretch}{1.2}
             \begin{tabular}{c}
               Text\\
               In A\\
               Circle\\
               Box
             \end{tabular}}
```



or even a `\psovalbox`:

```
\psovalbox[fillstyle=solid,%
           fillcolor=Orange,%
           linecolor=BrickRed]%
          {\color{SpringGreen}
           \large\bfseries
           \renewcommand{%
             \arraystretch}{1.2}
           \begin{tabular}{c}
                  Text\\
                  In An\\
                  Oval\\
                  Box
           \end{tabular}}
```

Another parameter for the various boxes is the `boxsep` whose default value is `true`. In this case, the box that is produced (in the TEXnical sense) is the size of the "frame" around it. If it is set to `false`, then the box produced is the size of what's inside, so that the frame is transparent to TEX. This is apparent only when the boxes are used within some surrounding text as illustrated below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to `tutorialteam@tug.org.in`

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
`http://www.tug.org.in`

```
\color{Blue}
Thus we find that $x+y=3$ and
using this together with
\psovalbox[linecolor=Red]%
          {$x^2+y^2=3$}
found earlier, we see that
$x=2$ and $y=1$

\vspace{1cm}

Thus we find that $x+y=3$ and
using this together with
\psovalbox[linecolor=Red,%
           boxsep=false]%
          {$x^2+y^2=3$}
found earlier, we see that
$x=2$ and $y=1$
```

Thus we nd that $x + y = 3$ and using this together with $x^2 + y^2 = 3$ found earlier, we see that $x = 2$ and $y = 1$

Thus we nd that $x + y = 3$ and using this together with $x^2 + y^2 = 3$ found earlier, we see that $x = 2$ and $y = 1$

Each of the boxing commands above has a *starred* version, which draws a *solid* shape around the enclosed text instead of just a frame. This is similar to the starred versions of graphic objects we've seen earlier, but the color of the boxes is determined by `fillcolor` instead of `linecolor` for other graphic objects.

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\SaveVerb{box}=\psframebox=
\psframebox[fillstyle=solid,%
            fillcolor=Cyan]%
            {\color{Red}
             \LARGE\bfseries
             Text In A \UseVerb{box}}

\vspace{1cm}
```

Text In A \psframebox

```
\SaveVerb{starbox}=\psframebox*=
\psframebox*[fillstyle=solid,%
             fillcolor=Cyan]%
            {\color{Red}
             \LARGE\bfseries
             Text In A \UseVerb{starbox}}
```

Text In A \psframebox*

(Here, the command pair `\SaveVerb` and `\UseVerb` come from the package fancyvrb and are used to get the control sequence strings `\psframebox` and `\psframebox*` as arguments of the commands.)

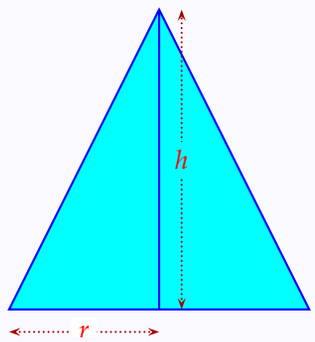
We've mentioned in Chapter 6 that the `\rput*` command puts the text first in a `\psframebox*`. But there are occasions when we've to use both `\rput` and `\psframebox` together instead of a single `\rput*`, as in the example below:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\psset{linecolor=Blue}
\begin{pspicture}(0,0)(4,5)
  \pspolygon[fillstyle=solid,%
             fillcolor=Cyan]%
            (0,0)(4,0)(2,4)
  \psline(2,0)(2,4)
  \psset{linecolor=Mahogany,%
         linestyle=dotted,%
         dotsep=1pt,%
         arrows=<->}
  \psline(2.3,0)(2.3,4)
  \rput(2.3,2){%
    \psframebox*[boxsep=false,%
                 fillcolor=Cyan]%
                {\color{Red} $h$}}
  \psline(0,-0.3)(2,-0.3)
  \rput*(1,-0.3){\color{Red} $r$}
\end{pspicture}
```



Note that here, we cannot use `\rput*` directly for the label h, since the default color of `\psframebox*` is white, (which is OK for the label r) but we want the color of the box for h to be cyan, to blend it with its background.

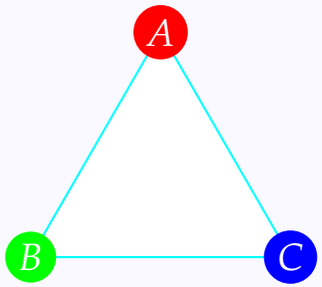
While on the topic of "putting", we should also mention the command `\cput` (and of course `\cput*`) which combines the functions of `\pscirclebox` and `\rput` (or `\rput*`), as shown in the next example:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\Large
\psset{fillstyle=solid,unit=2cm}
\begin{pspicture}(0,-1)(1,1)
  \SpecialCoor
  \pspolygon[linecolor=Cyan]%
            (0,1)(1;210)(1;330)
  \cput*[fillcolor=Red]%
        (0,1)
        {\color{White} $A$}
  \cput*[fillcolor=Green]%
        (1;210)
        {\color{White} $B$}
  \cput*[fillcolor=Blue]%
        (1;330)
        {\color{White} $C$}
\end{pspicture}
```



Online LaTeX Tutorial

Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

9.2. Clipping

We can *clip* text, that is, cut off everything outside a specified boundary, using the `\psclip....\endpsclip` commands. A simple example is given below:

```
\psclip{%
  \psdiamond[linecolor=Red,%
             fillstyle=solid,%
             fillcolor=Yellow]%
            (2,0.25)(2.5,0.5)}
  \color{Blue}\Huge Cut Diamond
\endpsclip
```

By careful use of coordinates, we can create an overlay effect with suitable clipping, as in the next example:

```
\begin{pspicture}(0,0)(6,2)
  \rput[bl](0.5,1){%
    \color{Blue}\Huge Cut Diamond}
  \psclip{\psdiamond*[linecolor=Yellow]%
                     (3,1.25)(2.25,0.5)}
    \rput[bl](0.5,1){%
      \color{OliveGreen}\Huge Cut Diamond}
  \endpsclip
\end{pspicture}
```

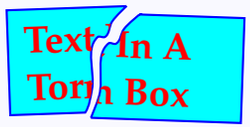
Or special effects like this:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(0,0)(2.5,2)
  \rput{2}{%
    \psclip{\pscustom[style=bluestyle]{%
              \psline(1.6,1.4)(0,1.4)(0,0)(1,0)
              \tearcurve}}
      \rput[bl](0,0){\color{Red}\Large\bfseries
        \begin{tabular}{c}
          Text In A\\
          Torn Box
        \end{tabular}}
    \endpsclip}
\end{pspicture}
\hspace{-2.6cm}
\begin{pspicture}(0,0)(2.5,2)
  \rput{-2}{%
    \psclip{\pscustom[style=bluestyle]{%
              \tearcurve
              \psline(1.6,1.4)(3,1.4)(3,0)(1,0)}}
      \rput[bl](0,0){\color{Red}\Large\bfseries
        \begin{tabular}{c}
          Text In A\\
          Torn Box
        \end{tabular}}
    \endpsclip}
\end{pspicture}
```



Here, the custom style `bluestyle` and the custom curve `\tearcurve` are defined as follows:

```
\newpsstyle{bluestyle}{%
  linecolor=Blue,%
  fillstyle=solid,%
  fillcolor=Cyan}
\newcommand{\tearcurve}{%
  \pscurve(1,0)(1.1,0.3)(1.2,1)(1.3,1)%
          (1.4,1.2)(1.6,1.6)}
```


Online LaTeX Tutorial
Part II – Graphics
PSTricks

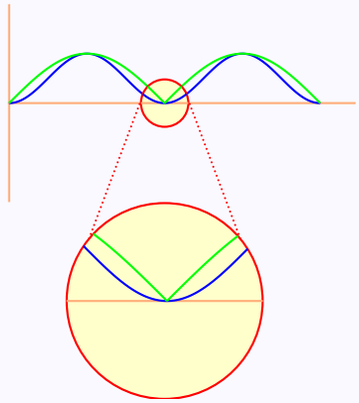
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

Incidentally,note that the `\psclip...\endpsclip` commands can be used to clip not only text, but graphic objects also, as shown in the example below:

```
\psset{unit=0.66,linecolor=Red}
\begin{pspicture}(0,-7)(6.5,2)
  \coloraxes(0,0)(0,-2)(7,2)
  \plotsqsin
  \plotabssin
  \psclip{\ycirc(3.1416,0){0.5}}
  \coloraxes(0,0)(0,-2)(7,2)
  \plotsqsin
  \plotabssin
  \endpsclip
  \dotline(2.6416,0)(1.1416,-4)
  \dotline(3.6416,0)(5.1416,-4)
  \psclip{\ycirc(3.1416,-4){2}}
  \psset{origin={3.1416,4},unit=1.33cm}
  \coloraxes(0,0)(0,-2)(7,2)
  \plotsqsin
  \plotabssin
  \endpsclip
\end{pspicture}
```



where the various customized commands used are as follows:

```
\newcommand{\plotsqsin}{%
  \psplot[plotpoints=500,plotstyle=curve,linecolor=Blue]%
         {0}{6.2832}{57.2958 x mul sin 2 exp}}
\newcommand{\plotabssin}{%
   \psplot[plotpoints=500,plotstyle=curve,linecolor=Green]%
          {0}{6.2832}{57.2958 x mul sin abs}}
\definecolor{PaleYellow}{cmyk}{0,0,0.2,0}
\newpsobject{ycirc}{pscircle}{fillstyle=solid,fillcolor=PaleYellow}
\newpsobject{dotline}{psline}{linestyle=dotted,dotsep=1pt}
\newpsobject{coloraxes}{psaxes}%
             {linestyle=solid,linecolor=Apricot,labels=none,ticks=none}
```


Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

9.3. Rotating and scaling

There are are also ready to use commands for rotating text left, right or down, leaving the needed amount of spaces.

```
go straight
\rotateright{\color{Red} go down}
\rotatedown{\color{Green} turn upside down}
\rotateleft{\color{Blue} go up}
go straight again
```

go straight go down turn upside down go up go straight again

A better effect can be obtained by computing the heights of various upright boxes and raising and lowering them by the appropriate lengths:

```
\newlength{\dlen}
\settoheight{\dlen}{%
  \rotateright{\color{Red} go down}}
\newlength{\ulen}
\settoheight{\ulen}{%
  \rotateleft{\color{Blue} go up}}
go straight
\raisebox{-\dlen}{%
  \rotateright{\color{Red} go down}
\rotatedown{%
  \color{Green} turn upside down}
\rotateleft{\color{Blue} go up}}
\raisebox{\ulen-\dlen}{%
   go straight again}
```

go straight go down turn upside down go up go straight again

For such manipulation of long pieces of text, these commands also have the "environmental" forms `\begin{Rotateleft}...\end{Rotateleft}` and others.

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\textbf{\color{Red} Question}:
\color{Blue} Why did the tachyon
cross the street?\\[10pt]
\begin{Rotatedown}
  \parbox{\linewidth}{%
    \textbf{\coor{Red} Answer}:
    \color{Blue} Because it's already
    on the other side}
\end{Rotatedown}
```

Question: Why did the tachyon cross the street?

Answer: Because it's already on the other side

Text can also be *scaled*, using the command `\scalebox`. The general form of this command is

`\scalebox`{*number1 number2*}{*text*}

where *number1* is the horizontal scaling and *number2* is the vertical scaling. If only one number is specified, it is used for scaling in both directions. This is illustrated in the examples below:

```
\scalebox{0.8 4}{%
  \color{Red}tall and lean}

\bigskip

\scalebox{4 0.8}{%
  \color{Green}short and fat}

\bigskip

\scalebox{2}{%
  \color{Blue}large but proportional}
```

tall and lean

short and fat

large but proportional

Using negative numbers for scaling, we can flip text around either axis:

Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TeX Users Group
This document is generated by pdfTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
{\Large\bfseries\color{Red} MIRROR}
 {\color{Blue}\rule[-0.3cm]{0.1cm}{1cm}}
 {\Large\bfseries\color{Red}
   \scalebox{-1 1}{MIRROR}}

\vspace{1cm}

{\Large\bfseries\color{Red} MIRROR}\\
{\color{Blue}\rule{2.5cm}{0.1cm}}\\
{\Large\bfseries\color{Red}
  \scalebox{1 -1}{MIRROR}}
```

MIRROR MIRROR

MIRROR

MIRROR

We also have the `\scaleboxto` command with the general form

`\scaleboxto(`*number1*,*number2*`){`*text*`}`

With this command *text* is scaled to have width *number1* units and height plus depth equal to *number2* units. If one of the numbers is set to 0, then the box s scaled to have width and height (plus depth) equal to the other number. (Of course, we cannot set *both* numbers equal to 0).

```
\scaleboxto(1.5,1){\color{Red}
   tall and lean}

\bigskip

\scaleboxto(7,0.2){\color{Green}
   short and fat}

\bigskip

\scaleboxto(3,0){\color{Blue}
   small but proportional}
```

tall and lean

short and fat

small but proportional

Online LaTeX Tutorial

Part II – Graphics

PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

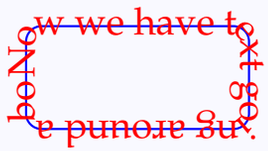
©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

9.4. Text along a path

One of the interesting features of the PostScript language is that it treats text as graphical object. This allows various manipulations of text. The package pst-text provides the command `\pstextpath` to set text along a specified path. Look at this example:

```
\begin{pspicture}(0,0)(3,2)
  \pstextpath{%
    \psframe[framearc=0.3,linecolor=Blue](0,0)(3,1.4)}{%
    \color{Red}\Large  Now we have text going around a box}
\end{pspicture}
```



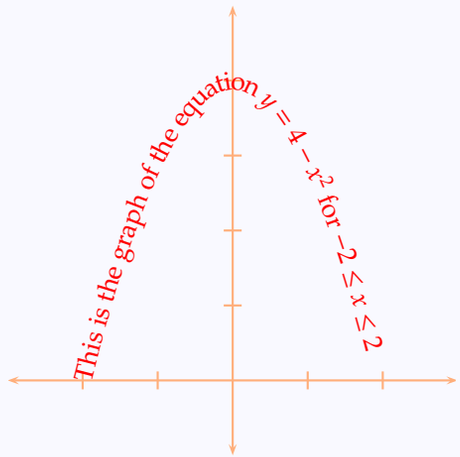
Note that the general form of the command `\pstextpath` is

`\pstextpath{`*graphic*`}{`*text*`}`

where, *graphic* specifies the path along which the specified *text* is to be set.

By default, `\pstextpath` draws also the graphic specified, but this can be suppressed by setting `linestyle=none`, as shown below:

```
\begin{pspicture}(-3,-1)(3,5)
  \colaxes[labels=none](0,0)(-3,-1)(3,5)
  \pstextpath{%
    \psplot[linestyle=none]{-2}{2}{4 x 2 exp sub}}{%
      \color{Red} This is the graph of the
                  equation $y=4-x^2$ for
                  $-2\le x\le 2$}
\end{pspicture}
```



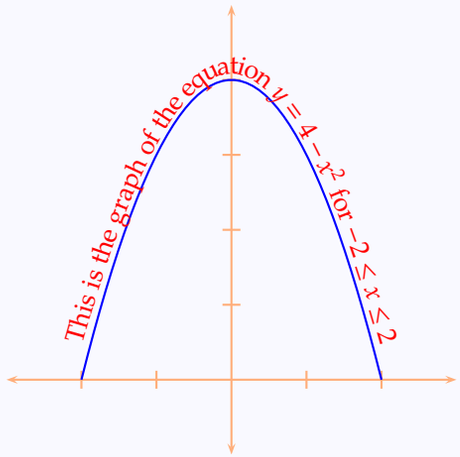
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TeX Users Group
This document is generated by pdfTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

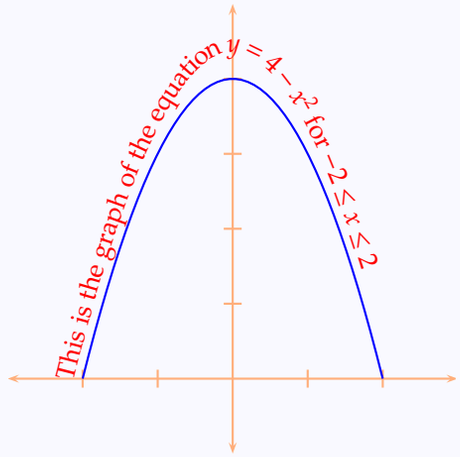
(Note that `\colaxes` used in he above example is a custom command, discussed in Chapter 7).

What if we need something like this?



The trick is to first draw the curve and then use `\pstextpath` to set the text along a slightly scaled up version of the curve, without actually drawing the second curve:

```
\begin{pspicture}(-3,-1)(3,5)
  \colaxes[labels=none](0,0)(-3,-1)(3,5)
  \psset{linecolor=Blue}
  \psplot{-2}{2}{4 x 2 exp sub}
  \psset{linestyle=none,unit=1.12cm}
  \pstextpath{%
    \psplot{-2}{2}{4 x 2 exp sub}}{%
    \color{Red} This is the graph of the
                equation $y=4-x^2$ for
                $-2\le x\le 2$}
\end{pspicture}
```



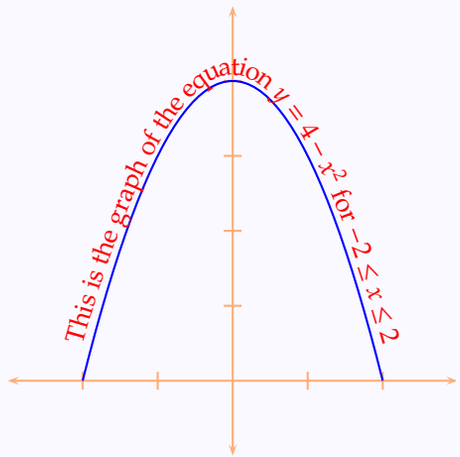
E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

But this is not exactly what we want. The trouble is that the command `\pstextpath`, by default, places the beginning of the text at the beginning of the path; however, it has an optional parameter which can be used to shift the position of the text:

```
\begin{pspicture}(-3,-1)(3,5)
  \colaxes[labels=none](0,0)(-3,-1)(3,5)
  \psset{linecolor=Blue}
  \psplot{-2}{2}{4 x 2 exp sub}
  \psset{linestyle=none,xunit=1.13cm,yunit=1.05cm}
  \pstextpath[c]{%
    \psplot{-2}{2}{4 x 2 exp sub}}{%
    \color{Red} This is the graph of the
                equation $y=4-x^2$ for
                $-2\le x\le 2$}
\end{pspicture}
```



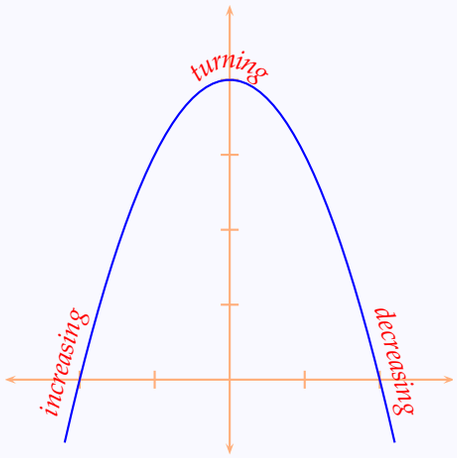
Note how we used the optional value `c` to center the text relative to the curve. (Note also the slight difference in `xunit` and `yunit` to get the text at the top just right). Other optional values are `l` (the default) for left justification and `r` for right justification. These are illustrated in the next example:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-3,-1)(3,5)
  \colaxes[labels=none](0,0)(-3,-1)(3,5)
  \psset{linecolor=Blue}
  \psplot{-2.2}{2.2}{4 x 2 exp sub}
  \psset{unit=1.15cm,linestyle=none}
  \pstextpath[l]{%
    \psplot{-2.1}{2.1}{4 x 2 exp sub}}{%
    \color{Red}\textit{increasing}}
  \pstextpath[r]{%
    \psplot{-2.1}{2.1}{4 x 2 exp sub}}{%
    \color{Red}\textit{decreasing}}
  \psset{unit=1.07cm}
  \pstextpath[c]{%
    \psplot{-2.1}{2.1}{4 x 2 exp sub}}{%
    \color{Red}\textit{turning}}
\end{pspicture}
```



Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

9.5. Text as graphic

The package pst-char provides the command `\pscharpath` which can be used to embellish text with colors and the like just as it were a graphic object. We give a couple of examples to illustrate this:

```
\begin{pspicture}(0,-1)(8,2)
  \DeclareFixedFont{\bigrm}{T1}{ptm}{m}{n}{1.5cm}
  \pscharpath[fillstyle=solid,%
              fillcolor=SkyBlue,%
              linecolor=Red]%
             {\bigrm PSTricks}
\end{pspicture}
```

PSTricks

Here, the command `\DeclareFixedFont` is the LaTeX way of specifying the font to be used.

```
\begin{pspicture}(0,-1)(8,2)
  \DeclareFixedFont{\bigsf}{T1}{phv}{b}{n}{1.5cm}
  \pscharpath[linecolor=Yellow,%
              fillstyle=gradient,%
              gradbegin=Yellow,%
              gradend=Red,%
              gradmidpoint=1,%
              gradangle=5]%
             {\bigsf PSTricks}
\end{pspicture}
```

PSTricks

This package also contains the command air `\pscharclip...\endpscharclip`, which like `\psclip...\endpsclip` pair, clips any object within them,but this time to the shape of the specified text:

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TeX Users Group
This document is generated by pdfTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\begin{pspicture}(-1.5,-1)(8,2)
  \DeclareFixedFont{\bigsf}{T1}{phv}{b}{n}{1.5cm}
  \DeclareFixedFont{\smallrm}{T1}{ptm}{m}{n}{2mm}
  \pscharclip[linestyle=none,%
              fillstyle=solid,%
              fillcolor=Cyan]%
             {\rput[bl](0,0){%
                 \bigsf PSTricks}}
     \rput[t]{90}(0,0){%
       \begin{minipage}{8cm}
         \offinterlineskip
         \newcounter{pscount}
         \setcounter{pscount}{500}
         \whiledo{\value{pscount}>0}{%
           \addtocounter{pscount}{-1}
           \color{Blue}\smallrm  Postcript}
       \end{minipage}}
  \endpscharclip
\end{pspicture}
```

(Here, the text to be clipped is "PostScript" written 500 times, in small font, specified by `\smallrm`, which is generated by the code starting with `\newcounter`, put in a box 8 centimeters wide and turned through ninety degrees.).

As in the case of `\psclip` this can also be used to produce an overlay effect.

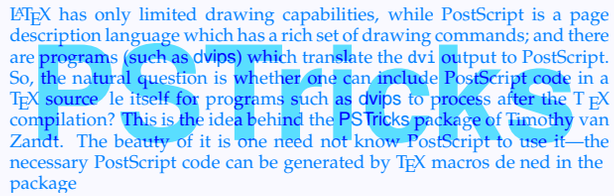
Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TEX Users Group
This document is generated by PDFTEX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TEX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

```
\DeclareFixedFont{\bigsf}{T1}{phv}{b}{n}{1.75cm}
\begin{pspicture}(0,-0.5)(8,3)
  \rput[bl](0,0){%
    \begin{minipage}{8cm}
      \color{RoyalBlue}
      \firstpara
    \end{minipage}}
  \pscharclip[linestyle=none,%
              fillstyle=solid,%
              fillcolor=CornflowerBlue]%
             {\rput[bl](0.25,0.15){%
                 \bigsf PSTricks}}
      \rput[bl](0,0){%
        \begin{minipage}{8cm}
          \color{Blue}
          \firstpara
        \end{minipage}}
  \endpscharclip
\end{pspicture}
```



Here, the command `\firstpara` is defined by

```
\newcommand{\firstpara}{%
  \scriptsize
  \LaTeX\ has only limited drawing capabilities, while
  PostScript is a page description language which has a rich set of
  drawing commands; and there are programs (such as \textsf{dvips})
  which translate the \texttt{dvi} output to PostScript. So, the
  natural question is whether one can include PostScript code in a
  \TeX\ source file itself for programs such as \textsf{dvips} to
  process after the \TeX\ compilation? This is the idea behind the
  \textsf{PSTricks} package of Timothy van Zandt. The beauty of it is
  one need not know PostScript to use it---the necessary PostScript
  code can be generated by \TeX\ macros defined in the package}
```


E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TeX Users Group
This document is generated by pdfTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in

just typesets the opening paragraph of our tutorial in `\tiny` font in an 8 centimeter wide box.

Online LaTeX Tutorial
Part II – Graphics
PSTricks

E Krishnan, CV Radhakrishnan and AJ Alex constitute the graphics tutorial team. Comments and suggestions may be mailed to tutorialteam@tug.org.in

©2004, The Indian TeX Users Group
This document is generated by pdfTeX with hyperref, pstricks, pdftricks and pdfscreen packages in an intel PC running GNU/LINUX and is released under LPPL

The Indian TeX Users Group
Floor III, SJP Buildings, Cotton Hills
Trivandrum 695014, INDIA
http://www.tug.org.in