11 Simple Diagrams in LaTeX

LaTeX provides support for simple diagrams in its `picture` environment. Consider diagram 7. This is generated by the source in figure 8.

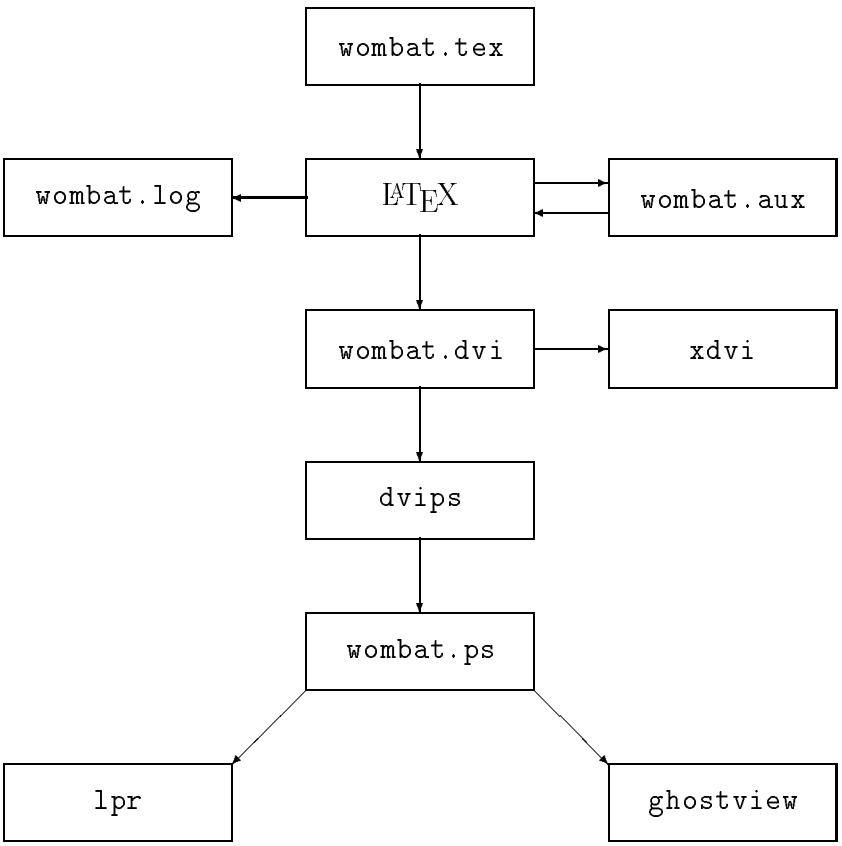


Table 7: An example diagram in LaTeX

```
%\unitlength is set in the preamble (to 1mm)
\begin{picture}(110,110)
% First row
\put(40,100){\framebox(30,10){\file{wombat.tex}}}
\put(55,100){\vector(0,-1){10}}
% Second row
\put(0,80){\framebox(30,10){\file{wombat.log}}}
\put(40,85){\vector(-1,0){10}}
\put(40,80){\framebox(30,10){\LaTeX}}
\put(70,87){\vector(1,0){10}}
\put(80,83){\vector(-1,0){10}}
\put(80,80){\framebox(30,10){\file{wombat.aux}}}
\put(55,80){\vector(0,-1){10}}
% Third row
\put(40,60){\framebox(30,10){\file{wombat.dvi}}}
\put(70,65){\vector(1,0){10}}
\put(80,60){\framebox(30,10){\cmd{xdvi}}}
\put(55,60){\vector(0,-1){10}}
% Fourth row
\put(40,40){\framebox(30,10){\cmd{dvips}}}
\put(55,40){\vector(0,-1){10}}
% Fifth row
\put(40,20){\framebox(30,10){\file{wombat.ps}}}
\put(40,20){\vector(-1,-1){10}}
\put(70,20){\vector(1,-1){10}}
% Sixth row
\put(0,0){\framebox(30,10){\cmd{lpr}}}
\put(80,0){\framebox(30,10){\cmd{ghostview}}}
\end{picture}
```

Table 8: The source for diagram 7

We will now consider this in detail as an illustration of how to create diagrams in LaTeX.

11.1 Scaling

All sizes and lengths in `picture` mode are given in terms of `\unitlength`. This is typically set in the preamble with a command like

```
\setlength{\unitlength}{10mm}
```

11.2 Size of picture

The `picture` environment is started with the command

`\begin{picture}`*(X,Y)*

The *(X,Y)* coordinate pair defines the size of the picture (in terms of the `\unitlength`). In all the subsequent examples, a box will be drawn to indicate the perimeter of the picture and to identify the origin (bottom left).

11.3 Placing objects in a diagram

The fundamental command of the `picture` environment is the `\put` command:

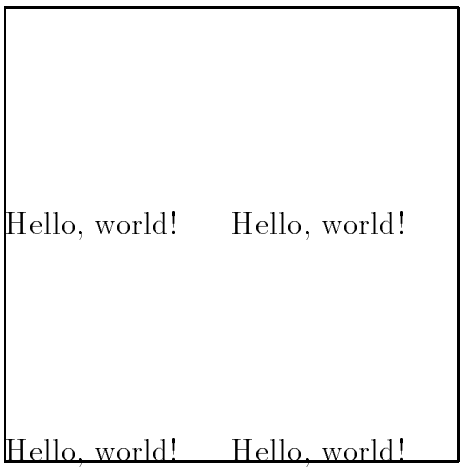
`\put`*(x,y)*`{`*object*`}`

- `\put` is the fundamental command of the `picture` environment.
- The *(x,y)* defines the location of the object being placed, with respect to the coordinate origin of the `picture` environment.
- The *object* is a "sub-picture" made from characters of a font designed for this environment.
- Because of this, there are a limited number of shapes and slopes available.

To place text, simply pass it as the graphical argument to `\put`. The coordinates in `\put` are the lower left corner of the text. For example:

```
\begin{picture}(60,60)
\put(0,0){Hello, world!}
\put(30,0){Hello, world!}
\put(0,30){Hello, world!}
\put(30,30){Hello, world!}
\end{picture}
```

gives[2]

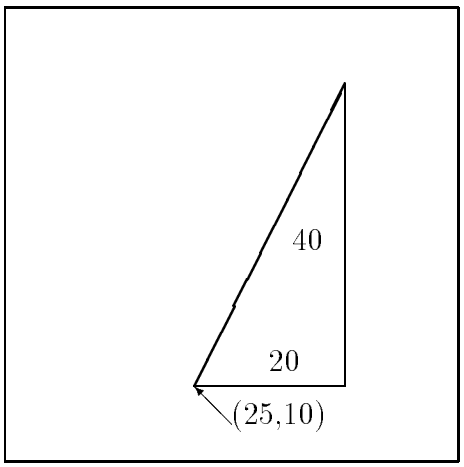


11.4 Lines and Arrows

Consider the command

```
\put(25,10){\line(1,2){20}}
```

This generates the line shown below in bold:



The things to note about this are:

[2]Remember that the frame is being added artificially.

- `\put(25,10)` determines where the line should start.
- `(1,2)` determines the *direction* of the line but *not* its length.
- `{20}` determines the extent of the line in the x-direction.

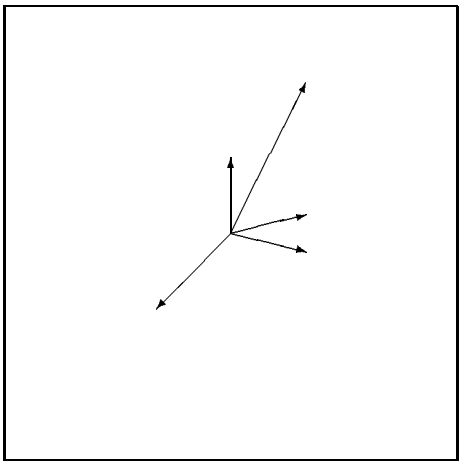
The line is constructed from a font of slopes, so only a discrete set of gradients is available. The rules for determining what slopes are available is as follows. Consider the *(x,y)* pair:

- x and y must lie in the range -6 to $+6$ inclusive.
- If either is zero, the other must be ± 1.
- Otherwise, x and y must be coprime.
- If x is 0 then the length parameter measures the y-direction.

There are 25 slopes available.

To get arrows, simply use the `\vector` command rather than the `\line` command. The slope numbers can only range from -4 to $+4$:

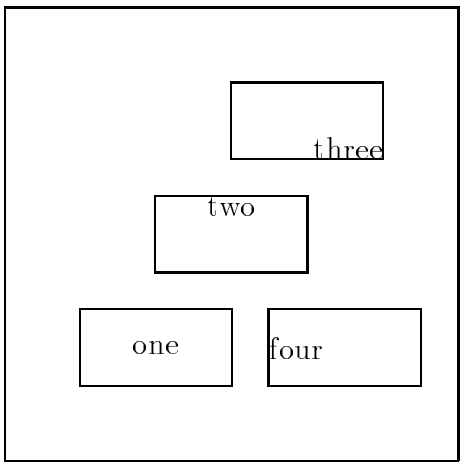
```
\begin{picture}(60,60)
\put(30,30){\vector(0,1){10}}
\put(30,30){\vector(1,2){10}}
\put(30,30){\vector(-1,-1){10}}
\put(30,30){\vector(4,-1){10}}
\end{picture}
```



11.5 Boxed Text

Putting a box around text is done by the `\framebox` command. `\framebox` takes a coordinate argument to determine the dimensions of the box, and an optional argument to control the justification of the text within the box:

```
\put(10,10){\framebox(20,10){one}}
\put(20,25){\framebox(20,10)[t]{two}}
\put(30,40){\framebox(20,10)[br]{three}}
\put(35,10){\framebox(20,10)[l]{four}}
```



11.6 Conclusion

The `picture` environment is vastly more powerful than has been described here. Refer to chapter 7 of Lamport's book for more details.