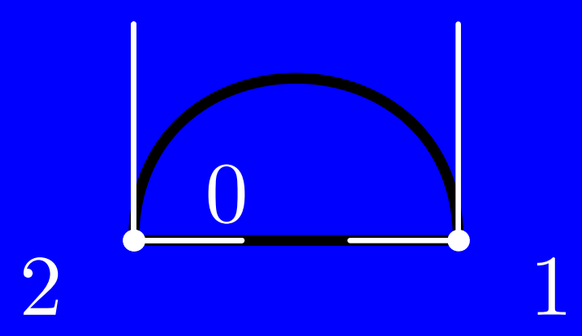
# METAPOSTEX

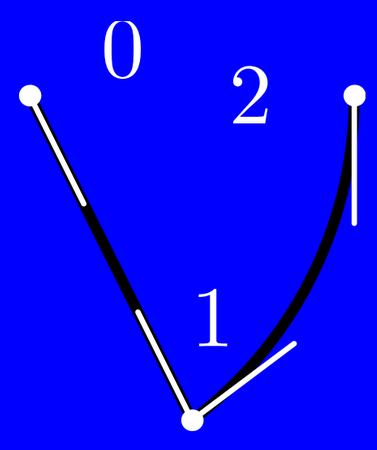

metapost-line.mp

```
beginfig(1);
draw (0,0) -- (30,0);
endfig;
end
```



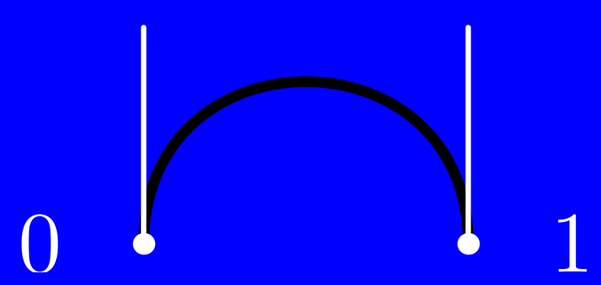

metapost-sun.mp

```
beginfig(1);
draw (0,0) ... (30,0){down} -- cycle;
endfig;
end
```



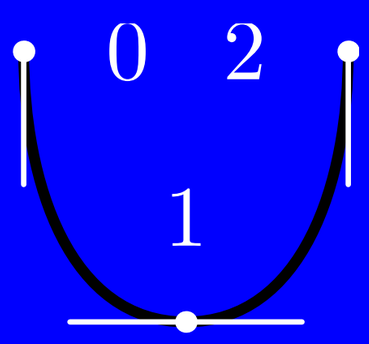

metapost-V.mp

```
beginfig(1);
draw (0,30) -- (15,0) ... (30,30){up};
endfig;
end
```



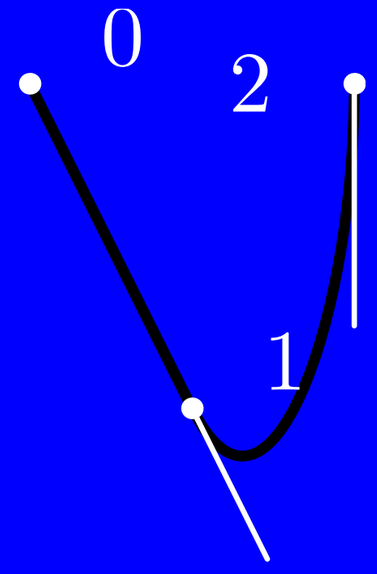

metapost-curve-up.mp

```
beginfig(1);
draw (0,0) ... (30,0)){down};
endfig;
end
```



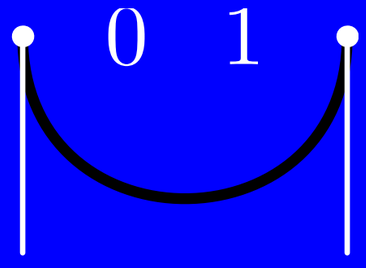

metapost-U.mp

```
beginfig(1);
draw (0,30){down} .. (15,5) .. (30,30){up};
endfig;
end
```



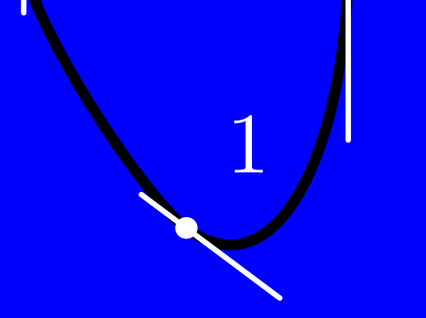

metapost-V.mp

```
beginfig(2);
draw (0,30) --- (15,0) .. (30,30){up};
endfig;
end
```



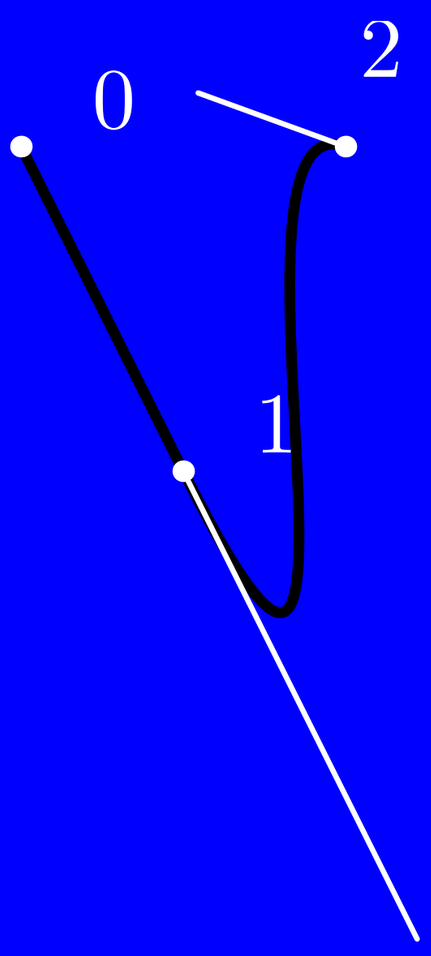

metapost-curve-down.mp

```
beginfig(1);
draw (0,0) ... (30,0){up};
endfig;
end
```



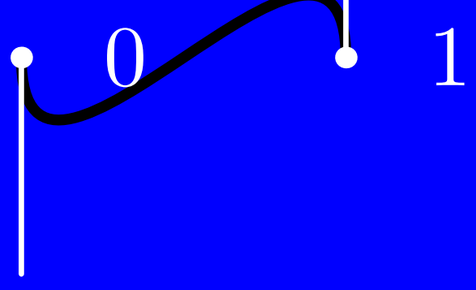

metapost-U.mp

```
beginfig(2);
draw (0,30){down} .. tension 2 .. (15,5)
... (30,30){up};
endfig;
end
```



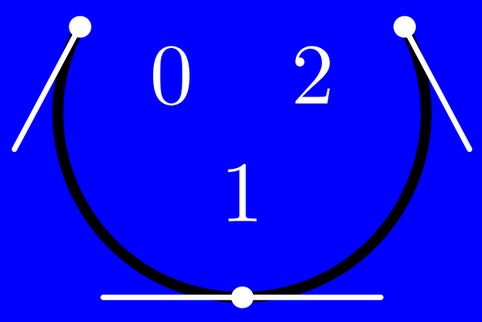

metapost-V.mp

```
beginfig(3);
draw (0,30) --- (15,0) .. (30,30){dir-20};
endfig;
end
```




metapost-wave.mp

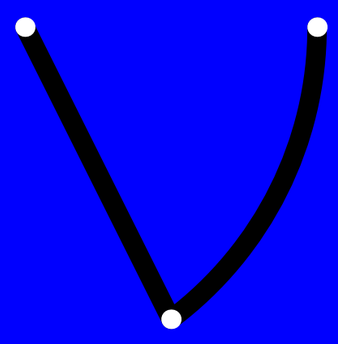
```
beginfig(1);
draw (0,0){down} ... (30,0){down};
endfig;
end
```

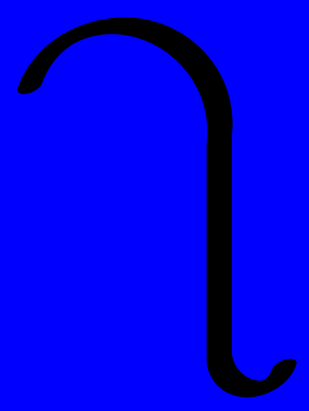



metapost-U.mp

```
beginfig(3);
draw (0,30) .. (15,5) .. (30,30);
endfig;
end
```

# METAPOSTEX





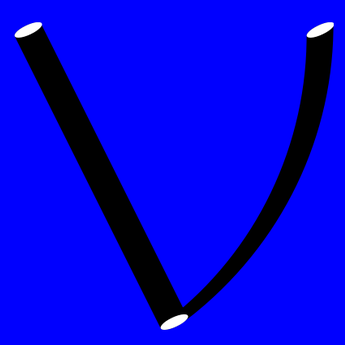
Parametric Font

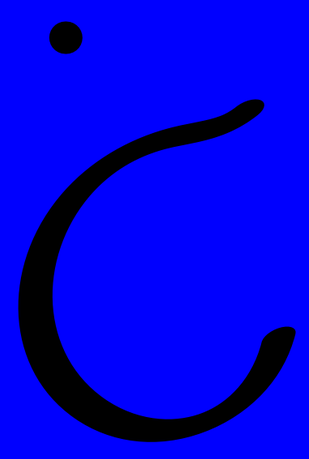
metapost-pen.mp

```
beginfig(6);
draw (0,30) -- (15,0) ... (30,30){up}
withpen pencircle scaled 2;
endfig;
end
```

draww.mp

```
beginfig(1);
pickup pencircle xscaled 2 yscaled 1
rotated 25 scaled 25
draw (0,900) .. (100,1000) .. (350,800) --
(350,400) .. (420,340) .. (470,380);
endfig;
end
```



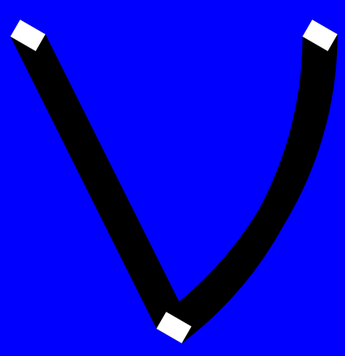


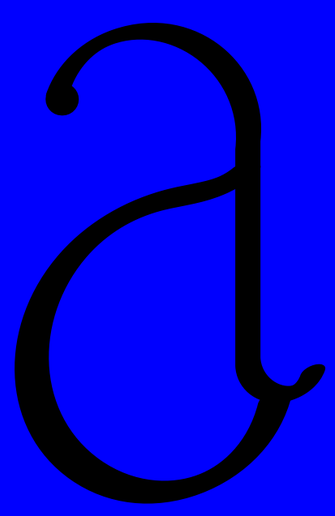
metapost-V.mp

```
beginfig(6);
draw (0,30) -- (15,0) ... (30,30){up}
withpen pencircle xscaled 3 yscaled 1
rotated 25;
endfig;
end
```

draww.mp

```
beginfig(2);
pickup pencircle xscaled 2 yscaled 1
rotated 25 scaled 34
draw (342,750) .. (300,725) .. (200,700)
.. (30,600) .. (50,200) .. (400,330);
pickup pencircle scaled 60
draw (6,885);
endfig;
end
```





metapost-V.mp

```
beginfig(6);
draw (0,30) -- (15,0) ... (30,30){up}
withpen pensquare xscaled 3 yscaled 2
rotated -30;
endfig;
end
```

metapost-V.mp

```
beginfig(6);
draw (0,30) -- (15,0) ... (30,30){up}
withpen penrazor xscaled 3 rotated 37;
endfig;
end
```

draww.mp

```
beginfig(3);
pickup pencircle xscaled 2 yscaled 1
rotated 25 scaled 25
draw (0,900) .. (100,1000) .. (350,800) --
(350,400) .. (420,340) .. (470,380);
pickup pencircle xscaled 2 yscaled 1
rotated 25 scaled 34
draw (342,750) .. (300,725) .. (200,700)
.. (30,600) .. (50,200) .. (400,330);
pickup pencircle scaled 60
draw (6,885);
endfig;
end
```

Note

Each metapost document must start with:

```
prologues := 3;
```

# METAMETAPOSTEX

This poster was initiated during Relearn Summer School, organised by OSP Open Source Publishing at Variable (Brussels) in the Summer 2013. It is a sequel to an open source poster made by OSP for the exhibition Visual Grammar at MAD Brussels in September 2012, showing the construction of Bézier and Spiro curves-based (type) design, laid-out with Inkscape.
The present poster is an attempt to have a similar approach with the principles of MetaPost and MetaFont, using LaTeX for the layout.
The title of the poster is a word play with the word TeX, which according to some users is supposed to be pronounced [ t ɛ ʁ ], "ter".

Despite the shifts in type design technologies, from wood and metal movable type to digital fonts, type designers almost always approach the font the same way: as a drawing and as outlines.

Nowadays, a digital typeface is usually drawn in vector softwares, which in terms of design experience means building and adjusting visual shapes (by moving points on Bézier curves) on a digital canvas. In parallel, in terms of computer programing, a series of programmatic instructions - code - is drawn.

In outline fonts, points define coordinates for the contour of the letter, its letterform and counterform (the circle inside an "o" for instance).

This font-design method defines the font by drawing its limits, its frontiers. This is a quite conventional way of working with fonts, dating from engraving wood type and cutting punches for metal fonts. And just like punchcutting a metal font, the font designer designs every letters and then every styles one by one. And while the designer draws his or her letter, the computer program transcripts the corresponding code.
Even though code is their essence, digital fonts today are hardly ever designed by writing code, except in a font and typesetting system created in 1979 by computer scientist and mathematician Donald Knuth: MetaFont. An early digital type system, MetaFont is an algebraic programming language to make stroke fonts.
At Relearn[1], the interest for stroke fonts systems like MetaFont was justified by their return to "gesture", calligraphy.
It is true that stroke fonts recall handwriting/carving, as both have as constructive element a center line along which the shape of the letter is "traced". Conceptually, stroke fonts are a return to gesture, for they propose to think and make fonts with their skeleton as a starting point. And the syntax used in MetaFont systems refers to the gesture of hand writing: "pick up pen", "draw"...
But in fact, very far away from any calligraphic hand or physical gesture (or a very specific one!), in this system the hand of the typographer basically hits buttons to write code.
It is though very close to the "digital gesture", that of the computer. It is writing letters with other letters: the alphabet and punctuation signs.
Programatic gesture.
Parametric
These programs were hardly ever reappropriated by graphic designers. Only a few recent examples show a resurgence of interest for that kind of way of approaching typefaces and design. OSP did a book using TEX in 2009. Dexter Sinister produced a MetaFont and wrote in 2010 a paper in their magazine Dot Dot Dot on the subject, "A Note on the Type" (Dexter Sinister, 2010). In 2011, Ecal published a book called Typeface as a Program, in collaboration with Jürg Lehni, designer, programmer and artist, who works on typography and programming. Finally, Simon Egli is now developing projects that make MetaFont technology accessible, or at least comprehensible, in the context of contemporary graphic design, through more visual interfaces, and an effort in translating and embedding passages from MetaFont to usual font formats et vice versa.
With the current tendency for self-reflexive, meta processes, this is quite in the air of time!

References:

Dave Crossland, *Why didn't METAFONT catch on?*, TUGboat, Volume 29 (2008), No. 3, 418-420.
http://ospublish.constantvzw.org/sources/vj10/
Dexter Sinister, *A Note on the*

[1]Summer School organised by OSP at Variable

*Type*, Dot Dot Dot 20 (2010), and in the first issue of Bulletins of the Serving Library (2011).

www.servinglibrary.org

www.metaflop.com

metapolator.com