Opening Pandora's Box

NEENIE BILLAWALA

841 Stendhal Lane
Cupertino, California 95014
sun!metamarks!nb

ABSTRACT

The motivation behind the Pandora project was to use METAFONT as a design tool rather than as a production tool in the creation of a typeface.

Pandora was developed by using generalized descriptions of the visual relationships between parts of characters, characters in a font, and fonts in a typeface family.

Pandora transforms from one font in her family to the next through different parameter settings applied to a single framework. It is not important that all variations look good, rather that a reasonable set can be found within the framework. A rich description allows a designer to quickly look at a number of possibilities.

1. METAFONT

1.1 Background

When I first heard about METAFONT in 1980, the idea seemed intriguing — using computers to design type. In 1984 when actually faced with the first generation of SUN computers, an experimental V operating system with a cryptic boot command, and version 0.** of METAFONT, I was not sure what it could really do — or rather, what I could convince it to do. The idea of designing type that was modifiable by the proverbial "turning of knobs" was seductive. And the idea that you, the designer, could choose those knobs was even more so.

It was not very confidence-inspiring to know that I had no computer experience and there was no manual yet to use as a guide. However, I did have some ideas of what I would like METAFONT to do and lots of help from the authors of the language.

Designing begins with ideas of how to solve a problem. What is the problem, what tools can be used to help solve the problem, how are those tools best used, what palette of solutions presents itself?

The problem of type design involves identifying the requirements of a design. Such requirements are outlined in a document called a *design brief*. For what purpose is the design being considered? Is it a text or a display face, or intended primarily for use as symbols? What visual flavor is it to have? What feeling does it convey? Will it have a single weight or will a family of weights need to be considered? Type is intended to be reproduced without deviation within a given range of tolerance. So, which technologies will be used in reproducing it? Additionally for METAFONT — what kind of flexibility do you want to build into the design to make it adaptable to different marking engines or different resolutions?

1.2 Drawing

Drawing type by hand is a special skill that takes years to develop, and even then, only a handful of people are successful at it. Having an eye and a sense of what belongs in a design takes another talent. Sometimes that talent combines in one person, but often it is the collaboration of these two types of skills that results in type.

Drawing type with METAFONT, a tool that can draw a precise line and a pleasing curve, takes time to learn also. Instead of the hand-eye coordination that a calligrapher or a type designer develops, you need to develop a feel for mind-eye coordination. You need to be able to translate visual shapes into the mathematical constructs that METAFONT understands. A steady hand is no longer needed as METAFONT will do the drawing for you. Though, an understanding of the problem and a sensitive eye is still required.

METAFONT uses pens to draw — and it has a broad notion of pens. These pens can have an arbitrary shape and size and inclination. They may even have no thickness, so that an outline can be filled. They relate to pens that we use for drawing, making the marks of a tip or nib that follows a path (Figure 1). However, the path traced out by the human hand will be much different from that traced out by a mathematical equation.

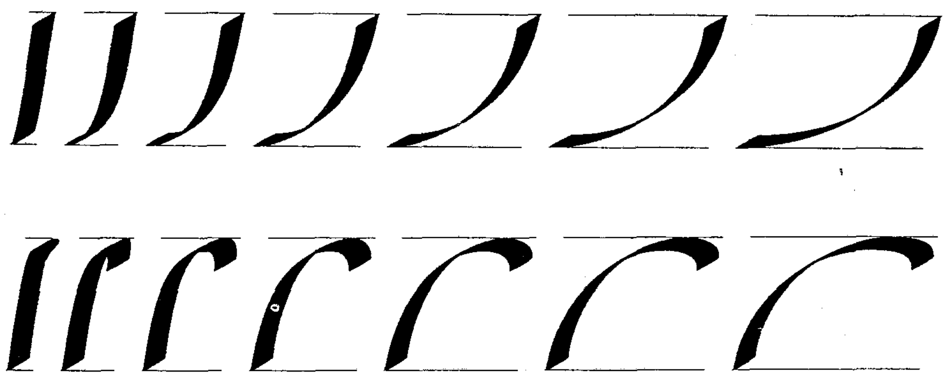


Figure 1: Pen strokes defined by two (top) or three (bottom) control points

A calligrapher's pen creates marks that are a combination of the shape of the nib, the consistency of the ink, the texture of the writing surface and subtle changes in pen pressure and pen angle. Such variety is difficult to recreate with METAFONT. While METAFONT can easily change pen nibs and pen angles, the "written" result is consistent — there is no fluctuation for writing surface or ink or pressure. This is not to say that one result is more or less beautiful than the other, just that it is not the same.

1.3 Designing

What makes METAFONT special is the fact that you can actually design with it. You can go through the same trial and refinement process of the traditional type design method — only the drawing tool looks a bit different.

METAFONT's strengths lie in its flexibility. The value of a flexible description becomes apparent as the technology changes. I remember a few years back when a 200 dpi XGP printer was being retired in the Computer Science department at Stanford. There were some 500 hand-tuned fonts that had been created for this printer and that represented man-years of work. Professor Art Samuel tackled the task of making these fonts work with the newer generation of 300 dpi laser printers. He developed an ingenious program that considered the pixel patterns, alternating rows of pixels and neighboring pixels, and then automatically converted a 200 dpi pattern to a 300 dpi pattern. In some ways the results were successful — previously created documents could still use the same fonts and they had about the same look as the 200 dpi printer. Unfortunately, the method could not take advantage of the higher resolution to improve the look of the fonts. Had those fonts been in a METAFONT format, it would have been a small matter to regenerate them at a higher resolution.

There are times when it might be useful to include more than resolution information into a typeface. You might have a printer where one-pixel lines are too thin. It might be a write-white printer or perhaps a high resolution imagesetter. It might be convenient to be able to set a minimum pixel thickness of 1 or 2 or even 5 as the resolutions increase. It might be useful to thicken up a design if you are going to print in reverse or increase all serif lengths by an arbitrary amount. Perhaps you just want a slightly different design. It might be that none of the above applies and other considerations will arise in the future.

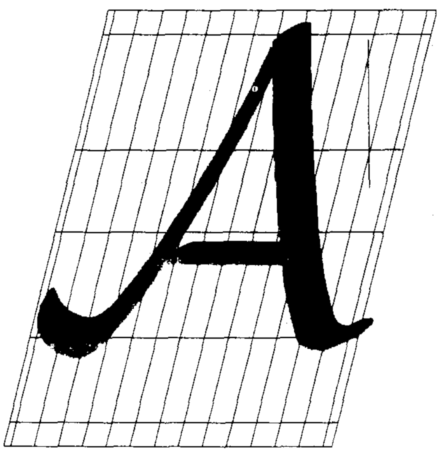
Once you learn how to transfer your ideas, you can embed more information into the character than could ever be guessed from a single drawing. Though it's possible to use METAFONT in a form of digital tracing, the potential richness of a flexible description would be lost.

METAFONT's value as a design tool is carried out by the fact that you can draw any number of iterations, changing one or many things in order to find a design solution. The drawing can be done more quickly than by hand, allowing the designer to consider many more possibilities and directions. With the aid of a printer to test variations and compare the results, you can draw and experiment.

2. Metamarks

In some ways type design lends itself well to description. The visual consistency that defines a typeface can be described in a program. Traits that are shared among characters — vertical dimensions, stem thicknesses, terminal treatments — can also share an analytical relationship.

The question is one of how to translate those visual relationships into language METAFONT understands. Words convey ideas, concepts and information well, but they are often lacking when it comes to describing something visual. It is really true that "one picture is worth a thousand words".



```
cmchar "Calligraphic A";
beginchar("A", 14.4u#, cap_height#, 0);
italcorr .5u#;
adjust_fit(-.05w#, 0); pickup cal.nib;
lft x5 = .54w; x7 = .9w;
top y5 = h + .4cap_curve; bot y7 = bot_flourish_line;
z6 = .3[z7, z5] - bend;
pickup tilted.nib;
lft x1 = .05w; x2 = .2w; rt x4 = x5;
y1 = y2 + .1h; bot y2 = bot_flourish_line; top y4 = h + .4cap_curve;
y3 = y6; z3 = whatever[z2, z4] + 2bend;
draw (z1 .. tension 1.2 .. {right}z2) softjoin flex(z2, z3, z4);     % left diagonal
pickup cal.nib;
erase fill (0, bot y5) -- (w, bot y5) -- (w, top y5) -- (0, top y5) -- cycle;
draw flex(z5, z6, z7) softjoin (z7 -- z7 + cal.extension);         % right diagonal
draw rt z3 -- z6;                                                  % bar
math_fit(.5u# - .1cap_height# * slant, ic#); labels(1, 2, 3, 4, 5, 6, 7); endchar;
```

Figure 2: Sample of METAFONT character output and source file (Computer Modern)

Mathematics has long been used as a tool to link the visual with the symbolic. METAFONT is an algebraic language. Algebraic equations describe the relationships you want to keep (Figure 2). Cubic Bézier curves are then drawn according to these relationships (Figure 3).

Certain relationships are very useful to maintain with type. The xheight of the typeface should be the same for all lowercase letters, just as the height of capitals should be the same. Sometimes all the numerals should have the same width so that they will align. With a drawn representation of a set of characters, measurements need to be taken to ensure consistency. A change to the xheight requires careful re-drawing of all the characters. However, should the xheight be a parameter that is shared by all characters, then a change to this parameter will have a global effect.

The first thing is to find a relationship in related parts of characters, and the next step is to look at a number of variations. It is a matter of experimentation and experience that will lead to discovery of those relationships that work well together. The visual consistency found in the characters of a typeface led to *Metamarks: Preliminary Studies for a Pandora's Box of Shapes*, an exploration of the related but not identical shapes that are found in several characters.

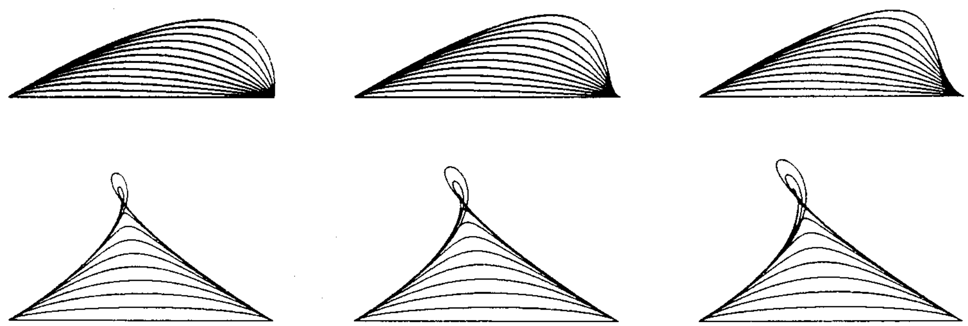


Figure 3: A series of Bézier curves, showing the general effect of control point placement

Part of the discovery of how METAFONT works lies in understanding its logic. You may understand how the curves are drawn and how parameters work, but you may still be surprised at the result. Since you can build a description, you must also be aware of the relationships you are building. You will need to think clearly about how the inter-relationships of the parameters will affect what you build.

Though everything is written down explicitly enough for a computer to understand, the specifications may not show the whole intent of the designer. It is virtually impossible to predict all outcomes for all cases, but with trials and error messages, METAFONT will direct you.

3. Pandora

Pandora started out as an experiment. Could this new tool and some wild ideas really combine into something that overlapped with the spirit of good type design? Was it possible to take the sense of traditional type design and apply it to the digital technology? Did the new format present options that were previously unavailable? Could the general requirements of functionality for type be satisfied? Generally, the answers turned out to be yes.

Pandora was the result of a *meta*-design brief that developed as I learned more about METAFONT, type and technology. Other than the idea that Pandora would be a family of roman-based forms and that text faces would be the main focus, there were no *specific* requirements to fulfill. No style had to be matched, no vertical dimensions maintained and no single marking engine was targeted. Pandora's was a meta-design brief, because many potential briefs were considered.

Mostly, Pandora is a family of text typefaces. I could see a relationship shared among text typefaces that was not as clear in display typefaces; for example, Bodoni and Helvetica are much closer than Calypso and Stop (Figure 4).

However, in place of the specifics that make up a traditional design brief, Pandora has parameters. Rather than have a specific xheight, Pandora has a parameter called *xheight*. Instead of a single stem weight, there are several parameters for setting stem thicknesses. The general slant of the typeface or obliqueness is also a changeable value. Pandora is special, because of her versatility.

3.1 Meta-Design

In meta-designing a typeface family, you need to think about the visual "essence" of a character. What makes a character recognizable? Is there an objective ideal shape? Do some cultures read letters better when they are of one style and others understand them better when they are of another? How much of what we consider to be clear and beautiful text comes from what we have grown up with? Is Univers really a universal typeface?

An essence cannot be defined by a single shape, but some of it is in every letter that we read. It is this indescribable essence that makes people from around the world understand the wide variation in handwriting and typefaces. Today we have a hard time reading the texts that were popular 500 years ago and printed in a blackletter type. Is it because we are not used to them or are they intrinsically harder to read?

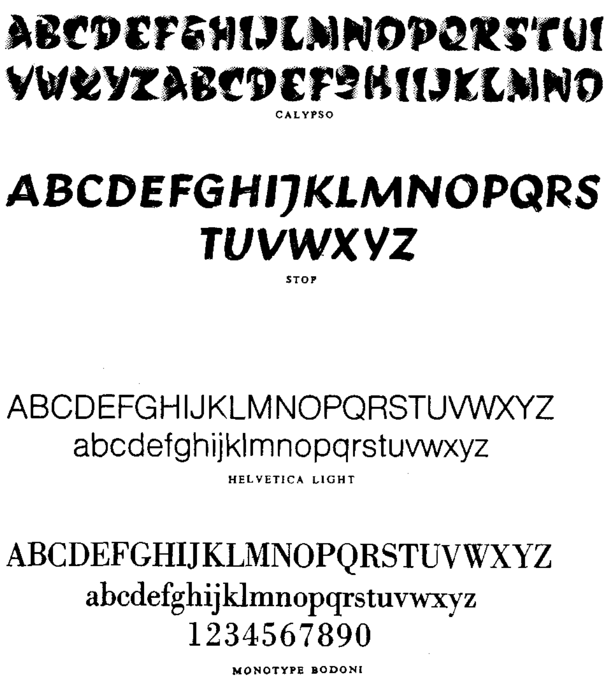


Figure 4: Contrast between display and text fonts

With Pandora I tried to capture some of this essence. I tried to construct the characters and the relationships that would preserve it. It is important to emphasize the features of each character that make it readily recognizable. Not that every variation would be a beautiful or appropriate solution, but that a set of reasonable solutions could be found. I did not look for the one and only solution, or an all-encompassing one — just a set that would work reasonably well and be flexible.

You can set relationships that should be preserved, and put in as much or as little control as you want. Sometimes it is interesting to give METAFONT a simple general guide and see what kinds of shapes can be created. It may lead you onto a path that you might not otherwise have considered. You might make a typeface that includes a certain amount of randomness. Or one that has several variations of the same character. It would be possible to create a set of three (or more) related interchangeable fonts and then perhaps set each third word with characters from each for the three fonts. Or you might want more control and place more conditions on your characters. For example, I have made the joining point of the arch in the lowercase *n* be on the stem somewhere between the top of any serif that there may be and the bottom part of the thickness of the arch. This avoids problems and fills the condition of most useful arches.

In looking for underlying relationships, does everyone think differently or is there something "objective" about the relationships? Separately, both Don Knuth and I came up with essentially the same serif scheme, though we never discussed it. On the other hand, there are many differences in the approach to Computer Modern and to Pandora. It would be interesting to see the results of giving the same set of characters to ten different designers, with the same instructions to fill out the family. Would there be ten different results? Would they all share some of the underlying concepts of change? In fact, Figure 5 shows the results of having 26 different designers create two characters (one uppercase, one lowercase), with the same vertical dimensions. However, it takes more than mathematical relations to make a uniform typeface.

The underlying concept will not apply to every kind of letter in every language. It is good judgement that decides the balance between a fair amount of flexibility and creating a separate basic form. Pandora and much of the Computer Modern family are good examples of roman forms. Other sets of basic forms will work for gothic characters, Chinese, Tamil and so on.

ABCDEFGHIJKKLM
NOPQRSTTUVWXYZ
abcdefghijklm
nopqrstuvwxyz

Figure 5: Results from 26 designers given only a few design parameters (Knuth 1984:106)

3.2 A Design Approach

Many of our text types come with a history whose origins begin with the written letter. It is a natural assumption when looking at a letter, to think that its inspiration might be a pen. Indeed, my first trials with METAFONT consisted of letters created by penstrokes. These characters tended to have a calligraphic flavor, but were somewhat limited.

However, type is created for a specific technology. Type does not exist without the means to create it. It is in the re-working of the forms from the hand-written inspiration that the deviations in shape occur — deviations that are no longer easily recreated by the penstroke.

For this reason, designing with penstrokes, whereby most of an unmodified METAFONT penstroke defines a character, can be limiting. It is useful in some cases and necessary in others. For the most part, however, I found it much more flexible to design with an outline, or a penstroke of no thickness.

The next step is to identify explicitly all those things that are important in the design. Vertical dimensions are an excellent example of this. You may know that you want lowercase characters to share an xheight, but you may not know what that value is to be. It may be between 50–55% of the point size, or it may be dependent on the height of capital letters, or it may be dependent on some other consideration. Rather than give an explicit xheight value in character description, you make the value of the xheight a parameter. Then as that parameter changes, all the places where the xheight is used will be changed.

The same applies for the height of capitals and all other shared vertical dimensions. When we talk about the height of a character, we are talking about the visual height of a character, not necessarily the mathematically measured value from top to bottom.

There are a few typical optical illusions that need to be considered. Look at the case where a circle, a square, and a triangle all share the same vertical and horizontal dimensions (Figure 6). The shapes all have a different visual weight. This is natural if you think about the area encompassed by each. Though vertical dimensions are equal, the circle and the triangle appear a bit shorter and narrower than the square. It shows that by at least one mathematical calculation, the "same value" does not result in the same visual result.

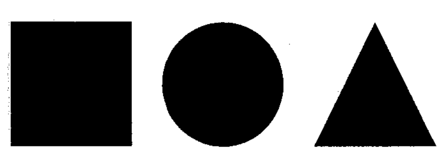


Figure 6: Optical illusion: square, circle and triangle with same vertical dimensions

These optical illusions carry over into type design. If the curved and pointed characters, such as an *O* and *A*, are to have the same visual vertical height as other characters, they may actually have to be a bit bigger in the vertical dimension. Those characters that have a strong horizontal feature, such as an *E* or a *T*, may have to be slightly shorter. One way to make the necessary adjustment is to build in another parameter that is dependent on the vertical height and perhaps on the resolution also.

Other features of a character combine to create a visual total. The inner shape or *counter* of a character plays a critical role. The proportion of thick and thin in a letter can combine to change the visual height and width. The example from *Metamarks* (Figure 7) gives a clear demonstration. Equal horizontal and vertical weights may give the illusion that one is heavier than the other. Unequal stem widths may actually look better. Typically weight is either added or sculpted away in order to achieve a pleasant and functional visual balance.

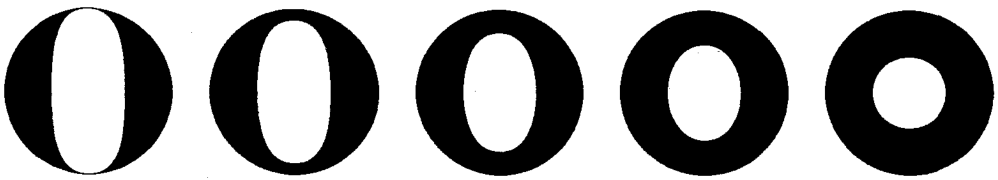


Figure 7: Circular shapes with variation in vertical stroke weight

Part of the design process remains the same. There are still the same problem solving aspects and visual considerations in the result. However, METAFONT does tend to change the approach. Rather than start with specifics, general parameters are built and visual relationships that have mathematical counterparts are constructed.

3.3 Parameters

Pandora relies on over one hundred parameters to define her shape. Some define dimensional limits, others relate to specific parts of characters, and some affect every character. The first group of parameters has values that are independent; the next group has values that are mostly dependent on the first group.

The independent parameters include common vertical values and thicknesses of stems and bowls. The *designsize* is set to the point size of the font. Other values, though independent, are implicitly dependent on the designsize or on each other. Typically the height of capitals or *cap* height is less than the designsize. An *xheight* is usually smaller than the *cap* height. Stem weights are usually less than bowl weights.

The dependent parameters are generally those that relate to parts of characters. These parameters are used in the macros that are then used to build the actual character descriptions. For example, there are a number of parameters that determine the shape of serifs and terminals. The *serif_thickness* is dependent on the xheight. It cannot be greater than half of the xheight, otherwise a character with two serifs would disappear. The *terminal_thickness* is related to the serif_thickness, and in Pandora they are set equal.

In addition to the serifs and terminals, there are parameters for arms, arches, bowls, circular shapes, junctures and notches. There are also settings for punctuation, accents, width, obliqueness, softness of the corners, and a number of other features.

However, the parameters are useless unless they have values. It is somewhat like working backward. You know you will need to have an xheight, but unlike a traditional method of working, you might not know how high it is until you have fit in the rest of the design. By defining the parameters, you will have necessarily thought about what each is meant to affect and you will have a starting point as to what value to give the parameters. In order to see an instance of the character or typeface, you will need to give an arbitrary value to each of the parameters. Not every possibility will work, but neither will every hand-drawn shape. It is more important to find a range of parameter settings that work together reasonably.

3.4 A Rich Description

Instructions for a letter to behave under certain conditions can be embedded into the character description. Algebraic equations specify conditions; booleans and conditional statements are used to make decisions. If all stems are to have the same number of pixels, then give them all the same value, and make sure they all begin on a pixel boundary. If point A is halfway between points B and C, METAFONT's way of saying this is `A=.5[B,C]`. In Pandora, there are alternate character possibilities for some of the lowercase letters. An upright style may use one variation, whereas an italicized version might use another (Figure 8).

Since the glue that ties the METAFONT family together is the underlying concept, this means that *fixed width* and *proportional* characters can share a relationship. Ideas of typeface consistency remain the same — stem widths are the same or similar, there is the notion of an xheight and cap height and so on. The major difference is that the widths have a special requirement that they all be the same. How do you fit the letter *m* into the same space as the letter *i*? Since letters were not meant to all have the same width, it is almost like trying to fit them into a straight-jacket. Variations can be made that might stand out in a normal text, such as adding or lengthening a serif in the *i* in a face that is otherwise sans serif.

Unlike previous technologies, there is no longer a need for different weights in a family to retain the same widths. Widths can vary according to the space they need to maintain the proper visual relationships. It may be more difficult to give a good description of what that relationship should be.

METAFONT also introduces a new feature not available with drawings — program bugs. They result from settings in the program that conflict with the way METAFONT draws. Sometimes the shape is affected, other times not. This is one of the trade-offs of having the ability to consider many possibilities.

Because you have to describe key relationships that are to be maintained, the METAFONT description gives more than a visual relationship. In fact, though related by METAFONT program and concept, the shapes of characters and the typefaces themselves may no longer be related in the traditional sense. METAFONT changes the meaning of a typeface family.

3.5 The Pandora Family of Typefaces

Pandora became an example of a flexible typeface family. It might only be the initiated who will recognize this, as the relationship is no longer purely visual. Many parameters or hooks have been built in and some have been used. Some of the parameters have limits placed on them, in the hopes of keeping the results reasonable. However, these too can be altered.

The inspiration for making a fixed-width typeface came out of a desire to print the METAFONT programs in Pandora and have the programs align as they did on the screen. It was a self-referential task — creating programs for characters so that the programs that created the characters could be printed.

The current family consists of 8 styles — serif and sans serif versions of a regular and a bold weight, a fixed pitch style, and slanted versions of each. The character sets match those of Computer Modern for the *typewriter* and *roman* styles. This means that the typefaces can be used interchangeably in terms of characters, though the look of the two families remains quite different.

4. Conclusion

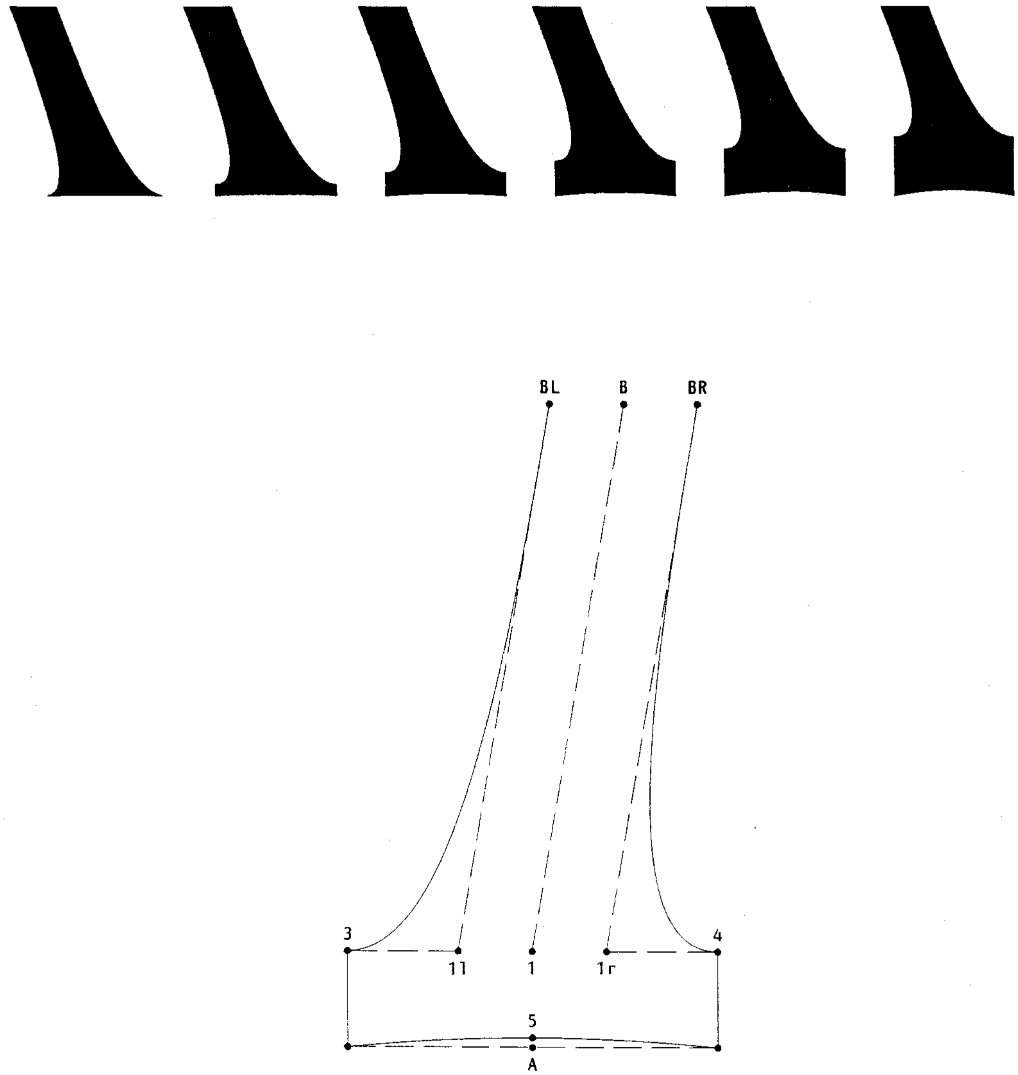
Tools like METAFONT make it very easy to generate and disseminate printable characters. No longer is a lengthy apprenticeship needed in order to have access to the tools that once were among a kingdom's most valued riches. Limited access had the advantage of a certain amount of quality control. The amateur was not likely to be in a position to do much damage, so to speak. On the other hand, accessibility spawns creativity and the sharing of information.

Bibliography

Billawala, Neenie. *Metamarks: Preliminary Studies for a Pandora's Box of Shapes.* Dept.of Computer Science, Stanford University. Report No.STAN-CS-89-1256.

Knuth, Donald E. "A Course on METAFONT Programming." *TUGboat* 5(2):105–118, 1984.

Knuth, Donald E. *Computer Modern Typefaces. Computers and Typesetting* Vol.E. Reading, Mass.: Addison-Wesley, 1986.

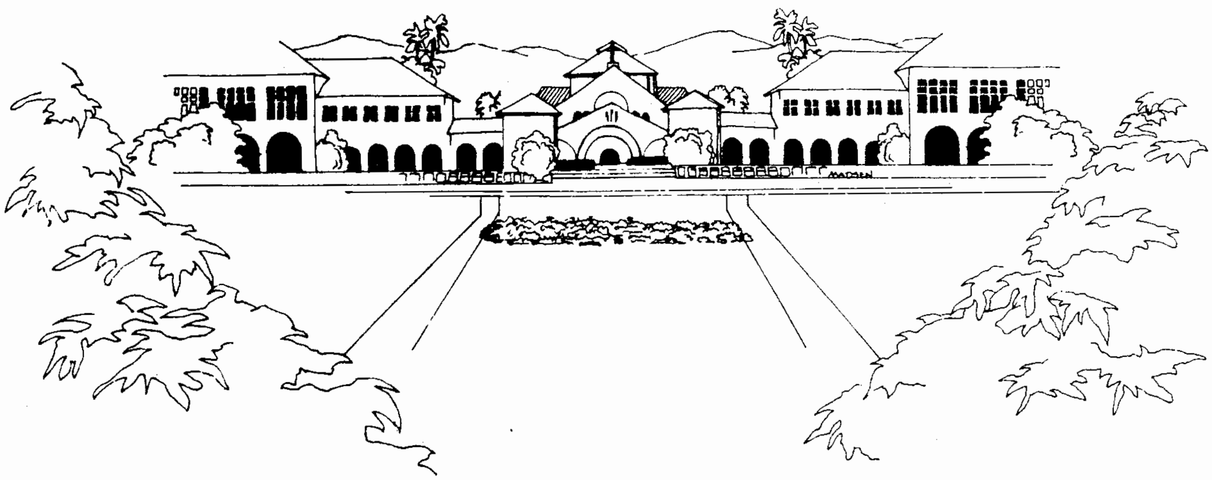


3.12 Serifs are parts of a character that often protrude from the stems of letters such as 'l', 'm', or 'n'. They may be very subtle, showing only as a slight bowing of a vertical stem; or they may be very prominent. The placement of a serif in a character usually has some historical relationship to how the character was drawn or written.

For our purposes, serifs have the following properties: They are always connected to a stem, even if that stem has no thickness. They have a left part and a right part, with both parts combining to form the full serif. They have a horizontal base, and a bracket area that connects the stem to the base. We may also take a small amount out of the bottom of the base — Matthew Carter has suggested that this be called "entasis." Midbracket pull is created by adding a point in the bracketed region through which the serif path must pass. This extra point lies somewhere on the line that protrudes from each bracket area. Bracketing can also be specified by choosing control points on the base and the stem.

The general serif illustrated here shows the key points constructed by the program on the opposite page.

Figure 8: Serifs: base and stem variations (top); general description (bottom)



AMS-TEX82 Users Course and TEX Users Group Meeting

Stanford University, July 11–15, 1983

Terman Engineering Center Auditorium