A page set in a Type 1 font

The three lines below are drawn from a font program in Adobe's first format. It is not a font file with tables to look a glyph up in: it is a PostScript program, its glyphs ciphered one by one, and the library reads it to find out which code reaches each of them. Each glyph is set as wide as the text metrics file that comes with the program says, or, without one, as the program itself says. The program is then carried into this file cut down to the glyphs this page draws, and reading software draws them from those very bytes.

Drawn from the program

HQF DEVELOPMENT

ABCDEFGHIJKLM NOPQRSTUVWXYZ 0123456789

A STROKE FACE, READ AND CUT TO SIZE.

What the program states about itself

It calls itself HqfStroke and holds 62 glyphs, of which 61 are reached by a code. Every glyph stands in one box, whose corners are 34 · -238 and 647 · 967; a vertical stem is 80 thousandths of an em thick, and an underline sits 120 below the line. Not one of those figures is written into this page: each is read out of the program.

What it weighs

The program itself is 8 117 bytes on disk, and a page setting the three lines above in it weighs 6 873 bytes, the program cut down to the glyphs those lines draw. Both figures are measured by the program that created this page, not written in by hand.

What travels with it

Like every other kind of font this library carries, a Type 1 program travels light: only the glyphs a page actually draws go into the file, so a page that says one word carries the letters of that word and not a whole alphabet. The format makes that a piece of work of its own: each glyph is ciphered on its own inside the program, so the library deciphers the program, rewrites its list of glyphs to hold only .notdef, the glyphs drawn and the glyphs an accented letter is built from, and ciphers it again. Each glyph kept is copied byte for byte, still under its own cipher, and the pieces of outline several glyphs share are kept whole. The font's description then names every glyph the file holds. On a face of a few dozen glyphs that saves little; on a large one it is most of the program: a face of sixty thousand bytes set in three words carries the glyphs of those three words and not the rest.

A Type 1 font is a simple font: its codes are one byte, so at most 256 of its glyphs are reachable at a time and it never sets text down the page. Which code reaches which glyph is the program's own business — this file repeats the program's own names for them.

The program, glyph by glyph

Its matrix is [0.001 0 0 0.001 0 0]; it slants 0°; each glyph has a width of its own; it spells out its own codes, 61 of them.

H is reached by code 72, the glyph H, which advances 680
Q is reached by code 81, the glyph Q, which advances 680
F is reached by code 70, the glyph F, which advances 680
2 is reached by code 50, the glyph two, which advances 680
0 is reached by code 48, the glyph zero, which advances 680
3 is reached by code 51, the glyph three, which advances 680
8 is reached by code 56, the glyph eight, which advances 680

HQF 2038

HQF 2038

The line above is set once from the widths the program states, and once with a metrics file written over it that widens every glyph by a quarter: at 15 points it runs 76.8 points, then 96.0.

Kerned from its metrics file

AT AV AW AY FA LT LV LW LY PA TA VA WA YA

AT AV AW AY FA LT LV LW LY PA TA VA WA YA

The same pairs twice: the first line is set without kerning, the second kerned on the pairs of the text metrics file the library reads beside the program. Each pair whose facing sides slant or stand open is brought together, by as much as a quarter of an em, and the move is written into the line itself, so every reader draws the letters where they were set.