

Massachusetts Institute of Technology

Artificial Intelligence Laboratory

**WORKING PAPER #78**

FED, the Font "EDitor"  
and  
Font Formats

by

Joseph D. Cohen and J. Pitts Jarvis

**Abstract**

This memo describes FED, a program used for compiling and inspecting fonts; AST font format, a text format which can be used to create and edit fonts; and KST font format, the binary format used by SCRIMP, TJS, and PUB.

Work reported herein was conducted at the Artificial Intelligence Laboratory, a Massachusetts Institute of Technology research program supported in part by the Advanced Research Projects Agency of the Department of Defense and monitored by the Office of Naval Research under Contact number N00014-70-A-0362-0005.

Working Papers are informal papers intended primarily for internal use.

FED, the Font "EDitor"

FED is a program used for compiling and inspecting fonts. FED understands the following font formats:

AST -- text format which can be used to create and edit fonts in TECO;

KST -- MITAI and CMU compiled font format, used by programs like SCRIMP, TJS, and PUB;

FNT -- SAIL compiled font format; and

XH -- Xerox PARC compiled font format.

FED reads all formats, but writes only AST and KST.

FED asks for an input file name. If the second name is AST, KST, FNT, or XH, FED assumes the input file to be in the appropriate format. If the second name is something else, FED asks what FORMAT the input file is in. FED reads the file and types out various decimal data. A FIXED width font is one in which the CHARACTER WIDTH of all defined characters is the same as the CHARACTER WIDTH of a space. FED calls all other fonts VARIABLE width.

FED tries to display the font on the user's terminal, if its a TV display, or on the 340, if the user is near there, and the 340 and PDP6 are available. FED displays as though one XGP dot equals one display dot, so the size and proportion of the font will be distorted. (200 horizontal XGP dots = 1 inch, 192 vertical dots = 1 inch. ~108 horizontal 340 dots = ~108 vertical 340 dots = 1 inch. ~58 horizontal TV display dots = 1 inch. ~57 vertical TV display dots = 1 inch.)

FED then offers to write out the font. If you type ^ (control X), FED kills itself; if you type N, FED restarts itself; if you type Y, it asks for the output file name; and if anything else is typed, it repeats its offer. The input default system name is FONTS, the output default system name is the job's SNAME. Input and output default second file name is KST.

Please do not write new fonts onto the FONTS directory -- it is almost full. Absolutely do not modify fonts on the FONTS directory. All concerned programs are perfectly capable of reading fonts from your very own directory. The reason for this injunction is that it has been determined, after much experimentation, that the definition of "improvement" is consistently a local variable. Please keep your "improvements" to yourself.

AST Font Format

The AST font format is a text format which can be used to create and edit fonts in TECO. An AST file has one page of font data, followed by one page of character data for each defined character. All numbers except CHARACTER CODE are decimal. It is the order of the numbers, rather than the text identification after them, that FED looks at. The identification is merely a convenience.

The file's first page is font data and looks like:

```
n KSTID
n HEIGHT
n BASE LINE
n COLUMN POSITION ADJUSTMENT
```

KSTID is a currently unimplemented feature. It is intended to be a unique number for each font so that PDP11 will know which fonts are loaded, and not do unnecessary reloads. KSTID is used by CMU. (KSTID should be set to 0.)

HEIGHT is the height of all characters in XGP raster lines.

BASE LINE is the location of the character base line in XGP raster lines from the top of all characters. (The base line is used by the XGP underscore command. See Working Paper #72, *XGP Font Catalog*.)

COLUMN POSITION ADJUSTMENT is an artifact. It is a number which is added to the LEFT KERN of each character to get its total left kern. To be SAIL compatible, COLUMN POSITION ADJUSTMENT should be 0, and all kerning included in individual character data.

The rest of the file is one page of character data for each defined character. Each page looks like:

```
n CHARACTER CODE
n RASTER WIDTH
n CHARACTER WIDTH
n LEFT KERN
```

```
* *
* *
*
* *
* *
```

CHARACTER CODE is the octal ASCII code of the character.

RASTER WIDTH is the width of the raster in XGP dots. It is a good idea to make the raster width as small as possible, in order to save space in font files, and to allow the XGP PDP11 to store more fonts. A leading blank column can be flushed if RASTER WIDTH and LEFT KERN are decremented. Trailing blank columns can be flushed by simply decrementing RASTER WIDTH. FED will have a feature implemented which does this automatically.

CHARACTER WIDTH is the amount, in XGP dots, that printing this character will advance the line position. This number is completely

independent of RASTER WIDTH, and LEFT KERN. For example, if CHARACTER WIDTH < RASTER WIDTH, characters will overlap. If CHARACTER WIDTH > RASTER WIDTH, there will be extra blank columns in the character. How many appear on each side is determined by LEFT KERN.

LEFT KERN (plus COLUMN POSITION ADJUSTMENT, if any) is the amount to move left from the logical left end of the character before printing the raster. (Positive left kern means move left, negative, move right.)

\* is any printing character used to define the CHARACTER RASTER. Each raster row is represented by a text line. If a space appears, the bit is off in the raster, if a printing character appears, the bit is on. Carriage return terminates the text row, and implies blanks for the rest of the raster row. Formfeed terminates the raster, and implies trailing blank raster lines. The first row is the top row. Information wider than RASTER WIDTH is ignored,

KST Font Format

KST is MITAI's and CMU's compiled binary font format. SCRIMP, TJ6, PUB, and other programs all use KST files as their font definitions. KST format is as follows:

Word 0/ KSTID

Word 1/ 9 bits COLUMN POSITION ADJUSTMENT, 9 bits BASE LINE,,HEIGHT

The remainder of the file has one block of data for each character. Each block looks like:

USER ID ;not currently used but low order bit is on. This bit can be used to find the beginning of a new block. The last block is followed by a -1.

LEFT KERN,,CHARACTER CODE

RASTER WIDTH,,CHARACTER WIDTH

CHARACTER MATRIX

The character matrix is stored four 8-bit bytes per word, so that ILOBs with 8-bit byte size get successive bytes. The remaining two low order bits are 0. The bits are reversed in each byte (because of the way the XGP interface works) so that the high order (leftmost) bit of the character is the low order bit of the byte. The matrix is stored row by row. Each row starts at a word boundary.

Odds and End

The FNT, XH, and FED internal formats are listed in the extravagantly commented FED source file (stored on the SYSENG directory). FED also has extensive comments on programming the PDP10 to PDP11 interface. FED was written by Pitts Jarvis, with additions by Joe Cohen. Brian Harvey is responsible for SAIL font hackery, whose specifications and fonts we copied.

THE END