The Program a2ac—A Font Handling on The PostScript Level.

Petr Olšák

Abstract. In this article, the `a2ac` program written by the author is described. The program enables to use PostScript fonts while typesetting texts in which accented letters are used. The font doesn't need to contain complete alphabet of a given language; the presence of mere accents themselves (not the whole accented characters) is sufficient. The configuration files of the `a2ac` program are independent on the PostScript font encoding as well as on the typesetting system encoding. The program may be used to prepare a font for any typesetting system, but only TEX was tested. The `a2ac` package is available on CTAN including the source in C and the documentation. It can serve as an alternative to the `fontinst` program by Alan Jeffrey. The program was tested on operation systems SUN OS, Linux and on DOS.

The program `a2ac` (Afm To Afm and add Composites) works with the `afm` file (Adobe Font Metrics) on its input and creates a new modified `afm` file on its output. The program reads so called "description file" during its work. The changes, we want to make, are defined in this file. The program adds the full description of new composites (accented letters) including kern pairs etc. to the output `afm` file.

The `afm` file describes an information about the letter using the symbolic name for each character (`Aacute` is the A with acute, for example). Each character is described through one of two possible ways. Either the character name is bonded to a definite encoding position and to a PostScript procedure for rendering the image of the character. Or the character is described as so called "composite character". In this case the encoding position of the character is set to be -1 and the description of how to make the character from elements is stored in `afm`. The elements are usually characters described as mentioned previously and only their symbolic names (not their encoding positions) are used.

All desired new composite characters are defined in the description file. Since only the symbolic names are used, the description file is totally independent on the encoding of the PostScript font and on the encoding used by the typesetting system.

The program has already nonzero intelligence when it reads the description file. You can declare and use so called "variables". You can write metric and composite information in form of simple expressions. You can add new kern pairs using the information from already existing similar kern pairs (with exceptions being possible).

An example of the description file format is shown in the appendix. In the example we define new letters É, Ě, é and ě. First, `Ecaron` and `Eacute` are defined as the composite letters (NC lines), then the new kerns are made for `Ecaron` and `Eacute`. The kerns are the same as the kerns for the letter E (NK line). Next, ecaron and eacute are defined as the composite letters with the same kern information as the letter e has (NC lines and NK line). Last, some exceptions for the kerns are specified. The pairs Pě, Tě and Vě will have no kern irrespective to the pairs Pe, Te and Ve, where the negative kerns are usually defined. The same exceptions are specified for the pairs Pé and Té.

For more information about the description file format see the a2ac-eng.doc file which is included to the a2ac package.

The ligature information is not defined in the description file. We will show that it essentially doesn't matter. Ligatures are described in the input afm file at the end of lines with the C prefix as is shown in our following example:

```
C 102 ; WX 333 ; N f ; B 20 0 383 683 ; \
              L i fi ; L l fl ;
```

The a2ac program just copies the ligature information to the output afm file. This is sufficient for generating the ligtables, which are used by the typesetting systems. For example, the afm2tfm program reads this information and generates the proper data in the ligtable of the tfm format. New ligatures for TEX can be defined through the enc files. For example:

```
% LIGKERN hyphen hyphen =: endash ; \
              endash hyphen =: emdash ;
```

in the file xl2.enc defines new ligatures for "--" and "---".

Font Preparation for TEX

After a2ac is used, preparation of the font for TEX goes on the standard way (as for the English language). You can use the afm2tfm program, which reads the converted afm file and the encoding information file enc to match the internal TEX encoding. The result is a TEX virtual font which includes two kinds of information: the information about reencoding from the internal TEX encoding to the raw PostScript font encoding (as was specified in the enc file) and the information about building the accented letters from the elements (as was specified in the "rebuilded" afm file).

Let us show an example of the PostScript font preparation for TEX for Czech and Slovak languages. First, we run the a2ac program. The command line looks like:

```
a2ac input.afm desc.tab output.afm
```

The first parameter is the name of the input afm file, the second one is the name of the description file and the third one is the name of the output `afm` file. The extensions (`.afm`, `.tab`) are obligatory—the program has no algorithm for adding the extensions automatically.

The command to run the `a2ac` program is usually a part of a script or a batch file. For example, the UNIX script to convert the font from the `afm` format to the `tfm` format may look like:

```
a2ac $1.afm cscorr.tab c$2.afm
afm2tfm c$2.afm -t xl2.enc -v c$2 r$2
vptovf c$2.vpl c$2.vf c$2.tfm
```

The name of this script is `mkfnt`, say. Then the usage is simple:

```
mkfnt Times-Roman ptmr
```

The file `cscorr.tab` is included in the `a2ac` package. The file includes definitions of composites for Czech and Slovak accented letters (i.e. the output font metric includes Czech and Slovak letters and the proper kerns for these letters). The input font has to contain the standard set of 26 latin letters (lower- and upper-cases) and all the accents necessary for the composite letters, which means the acute, caron, ring, quoteright, dieresis, circumflex accents. If the Adobe font in StandardEncoding is used then this requirement is satisfied. The output `afm` file then specifies all characters (plain or composite) of the Czech and Slovak alphabets and the new kern information.

The description file `cscorr.tab` was used without changes for all standard PostScript fonts (32 fonts installed in all RIPs) with a very good result. Sure, if you want, you can rearrange the `cscorr.tab` file for your new font to obtain the best result.

If the UNIX script above is used, the TeX font suitable for Czech and Slovak texts is prepared in the $\mathcal{CS}$-font encoding.[1] The output of the script is `c*.tfm` (for TeX input), `r*.tfm` and `c*.vf` (for `dvips` input). The only work, we must do, is to place these files to the appropriate directories (or to rearrange the script to make this work automatically) and to add one line to the configuration file `psfonts.map` of the `dvips` driver. The line looks like:

```
rfont PostScript-Font
```

If the PostScript font is not installed in the PostScript RIP of the output device, we have to store the `pfb` (`pfa`) format of the PostScript font in

[1] The $\mathcal{CS}$-font encoding is an extension of the Computer Modern font encoding. See paragraph "The TeX Encoding".

our computer. Then the line in `psfont.map` looks like:

```
rfont PostScript-Font </path/to/file.pfb
```

or (for DOS):

```
rfont PostScript-Font <d:\path\to\file.pfb
```

We can load the new font into the TeX using commands:

```
\font\newfont=cfont
{\newfont the eight-bit text in the same
          encoding as defined in xl2.enc
          can be used
          and the new font will be shown.}
```

The text in the new font will be printed when it is TeX-ed (and `dvips`-ed).

The TeX Encoding

The input text encoding for TeX doesn't need to be the same as the internal TeX encoding. The re-encoding can be performed by the `tcp` table of the emTeX, by changes of the `xord`/`xchr` vectors in the `tex.ch` file (the TeX's WEB source hacking) or by setting active characters. The `enc` file describes exactly the same encoding as the internal TeX encoding in which the hyphenation patterns have been read.

In the Czech Republic and in the Slovak Republic, so called CS-encoding (the encoding of the CS-fonts) is used very often as the internal TeX encoding. The CS-encoding is an extension of the Computer Modern text font encoding and follows the ISO-8859-2 layout for the Czech and Slovak letters. The CS-fonts are included in the widespread TeX installation package with the name CSTeX.

The advantages of the CS-encoding (in comparison to the T1 encoding) from our point of view are following:

- The fonts include the letters from Czech and Slovak alphabets only. The font generating from the METAFONT sources is faster and the `pk` files are smaller. Exceptions (non Czech/Slovak letters in names, for example) are accessible by plainTeX macros or by `\accent` primitive.
- The CS-font encoding is the conservative extension of the Computer Modern text fonts. Therefore, CS-fonts can replace CM text fonts in the installation packages without problems with the backward compatibility. This feature saves a space on the disk. No complicated macro re-definitions are necessary. For example, the `csplain` format, which is included

in $\mathcal{C}\mathcal{S}$TEX package, gives the same results as the original plainTEX. There are only three differences: The new letters with the code ≥ 128 are introduced by `\catcode`, `\lccode`, `\uccode`; $\mathcal{C}\mathcal{S}$-fonts are loaded instead of the CM text fonts; the Czech and Slovak hyphenation patterns alongside the US English are read.

• The layout of the accented letters in the $\mathcal{C}\mathcal{S}$-font is the subset of the ISO-8859-2 standard. Therefore, it is possible to use an eight-bit input Czech/Slovak text without re-encoding in the UNIX installations. The power is in simplicity! The `tcp` table (emTEX) is used in the DOS installation because there are many different layouts for the Czech/Slovak alphabet and there is no fixed standard in DOS (in contrast to the UNIX-like systems).

• Kerning and other aspects of the fonts are language dependent. The multi-language font (T1 encoded, for example) is a nonsense from the typhographer's point of view. In addition, we can't rely on the results of the font hacking from abroad.

The Computer Modern text font encoding (and its conservative extensions) is font-type dependable. For example, two different characters are encoded to the same position depending on `\rm` versus `\tt` font (fi versus downarrow for example) or on `\rm` versus `\it` font (the dollar versus the sterling). This feature implies some little problems.

Two files, `xl2.enc` and `xt2.enc`, are included in the `a2ac` package. These files define extensions of the $\mathcal{C}\mathcal{S}$-font encoding. The extensions include some characters which are not contained in the Czech and Slovak alphabets (i.e. in the $\mathcal{C}\mathcal{S}$-font), but which are contained in the Adobe StandardEncoding material.

It is recommended to use `xl2.enc` for `\rm` and `\it`-like fonts and `xt2.enc` for `\tt`-like fonts. The dollar is on the position 36 under any circumstances and the position number 132 was given to the sterling. If you load the PostScript `\it`-like font then you need to know that the sterling is not accessible by the dirty plainTEX trick `{\it\$}` but you need to use a macro with `\char132` code.

The position number 32 is not defined in the `xl2.enc` file because the cross accent for the Polish L and l is not included in the Adobe StandardEncoding. The Ł and ł themselves are on the positions 163 and 179 respectively. If we need to use these characters in plainTEX with the PostScript font, the re-definition of the macros `\l` and `\L` is necessary.

Note that the three-letter ligatures are not included in the Adobe StandardEncoding. This doesn't cause any problems because the three-letter ligatures are not used in Czech and Slovak. If the font is available in the ExpertEncoding you can do some hack on the `vpl` level to obtain these ligatures.

If you want to typeset the math by the PostScript font, you need to edit the `vpl` file [1] to include the uppercase Greek characters taken usually from the Symbol font. Sorry, but it is not obvious how to typeset the math by the PostScript font; some more hack must be done: on the plainTEX macro level as well as on the `\fontdimen` font information. The pretty "look" of math is a result of a font designer's work and a large TEX-macro programming. "Its emphasis is on art and technology, as in the underlying Greek word." (see [2], chapter 1). Therefore, the pretty math is not accesible from the text PostScript font via any automatic process.

For more information about the CS-font encoding and its history see [4] and the appendix F in [3].

About The Names of The Accented Letters

The names `dquoteright` and `tquoteright` should be used for the characters d' and t' in the description file `cscorr.tab`. We use the names `dcaron` and `tcaron` instead. The reason is, the characters d' and t' have their uppercase alternatives Ď and Ť, i.e. `Dcaron` and `Tcaron` respectively. The parameter `-V` (for `afm2tfm` to make the small caps variant of a font) does not work with the names `dquoteright` and `tqouteright`. We use `Lcaron` and `lcaron` instead of `Lquoteright` and `lquotering` because the semantic for these accents is same as for `Dcaron`, `dcaron`, `Tcaron` and `tcaron`.

References

[1] Donald Knuth: Virtual fonts, a more fun for grand wizards. TUGboat 11(1):13–23, April 1990.

[2] Donald Knuth: The TEXbook, vol. A of Computers & Typesetting. Addison-Wesley, Reading, MA, USA, 1986.

[3] Petr Olšák: Typografický sytém TEX (Typesetting System TEX—written in Czech), CSTUG 1995, 270 pages. ISBN 80-901950-0-8.

[4] Petr Olšák: Úvaha o fontech v CSTEXu (A Reflection about fonts in CSTEX—written in Czech), TeXbulletin (of CSTUG) 3/93 (121–131).

Appendix: A Part of The Description File.

```
NC Ecaron 2 ; PCC E 0 0 ; PAT caron 0 Carontop ;
```

```
NC Eacute 2 ; PCC E 0 0 ; PAT acute Acuteshift Acutetop ;
NK (Ecaron,Eacute) : E
NC ecaron 2 ; PCC e 0 0 ; PAC caron 0 0 ;
NC eacute 2 ; PCC e 0 0 ; PAC acute acuteshift vshift ;
NK (ecaron,eacute) : e
RK (P,T,V) ecaron 0
RK (P,T) eacute 0
```