NAME

cmap2enc − convert glyph indices in a TrueType-flavored OpenType font to Adobe glyph names

SYNOPSIS

cmap2enc [options] *font encoding*

DESCRIPTION

Many TrueType-flavored OpenType fonts contain no glyph names. When creating an *afm* file with *ttf2afm*, this causes all glyphs to be named something like `index0x01E7`, which makes the *afm* file almost useless for use with *fontinst* or *afm2tfm*.

cmap2enc uses the font's 'cmap' table to map glyph indices to Unicode values, and Adobe's *glyphlist.txt* file to map these to glyph names.

cmap2enc creates a new encoding file (for downloading with the font file) and a *mtx* file (for use with *fontinst*). The encoding file contains the same glyphs as the original one, but has all glyph names replaced with the corresponding glyph indices; the *mtx* file maps these glyph indices back to the original glyph names (it consists of many commands of the form `\renameGlyph{index0x01E7}{Euro}`, plus a definition of that `\renameGlyph` command).

Some of the glyph substitution features of OpenType fonts are supported; e.g. when the 'smcp' (Small Caps) feature is specified, the glyph names of the lowercase glyphs will be mapped to the glyph indices of the small capitals.

A list of features supported by the font can be generated using the **−p** option, see below.

EXAMPLES

NOTE: the *examples/* directory in the *fontools* distribution contains several complete examples of the use of **cmap2enc** and other tools from the *fontools* collection.

- To display all scripts, languages and features supported by Linotype Palatino:

```
cmap2enc -p pala.ttf
```

- To install Linotype Palatino in T1 encoding, with oldstyle figures:

  1. Generate an encoding file *t1j-pala.enc* and a metrics file *resetpala-t1j.mtx*:

```
cmap2enc -f onum pala.ttf t1.enc
```

  2. Create an *afm* file for the font; this is done best using *font2afm* (also in the *fontools* collection):

```
font2afm pala.ttf
```

  3. Reencode the *afm* file using the *t1j-pala.enc* encoding to create a 'raw' font *rpplrj9d*, with either *afm2afm* (also in the *fontools* collection) or *fontinst* (but be warned that *fontinst* might take some time, and can even crash if the *afm* is really big):

```
afm2afm -e t1j-pala.enc -o rpplrj9d.afm pala.afm
```

     or

```
\transformfont{rpplrj9d}{\reencodefont{t1j-pala}{\fromafm{pala}}}
```

  4. Create a 'value-added' *vpl* file from this raw font by adding ligatures, faking missing glyphs etc. with *fontinst*:

```
\installfont{pplrj9d}{rpplrj9d,resetpala-t1j,newlatin}%
    {t1}{T1}{pplj}{m}{n}{}
```

OPTIONS

cmap2enc supports the following command-line options:

−f *feature*[,*feature*]*
: Use the specified features when matching glyph indices to glyph names. Multiple features can be specified as a comma-separated list, **without** spaces between the items.

−l *language*
: Select the specified language. The default is 'DFLT' (surprise!).

−p Print some info about the font, plus a list of all scripts, languages and features contained in this font's GSUB table; then exit. (When using this option, the *encoding* need not be specified.)

−s *script*
: Select the specified script. The default is 'latn'.

−x *extra*
: Use the user-specified substitutions from *extra*.

FILES

All input files are searched using **kpsewhich** or **findtexmf** when these commands are available; otherwise only the current working directory is searched.

font
: The font file can be any OpenType font, though **cmap2enc** is probably only useful for the TrueType-flavored variant (PostScript-flavored ones always contain glyph names in the 'CFF' table).

encoding
: The encoding files read and written by **cmap2enc** are in standard *dvips* format. The name of the output encoding file is `<encoding><suffix>-<font>.enc`,

  where

  encoding is the name of the input encoding file (without the extension '.enc');

  suffix is `c` if the 'smcp' feature was selected, `d` for 'titl', `j` for 'onum', `w` for 'swsh', `0` for 'subs' and `1` for 'sups'; if multiple of these features were selected, their suffixes are concatenated in the order shown here;

  font is the name of the input font file, without the '.ttf' or '.otf' extension.

mtx The output *mtx* file is in standard *fontinst* format. The name of this file is `reset<font>-<encoding><suffix>.mtx`,

  where *font*, *encoding* and *suffix* have the same meaning as above.

extra
: The option **−x** can be used to name a file containing extra substitutions that complement or override the ones specified by the selected features. It consists of one substitution per line; each is either of the form `<from>;<to>`, where *from* and *to* are both glyph indices (specified as four hexadecimal digits), or of the form `<glyph name>;<glyph index>`, where *glyph name* is any glyph name and *glyph index* the glyph index (again, four hex digits) that is to be mapped to the specified glyph name.

RESTRICTIONS

- Most OpenType fonts contain several tables in the 'cmap' table, for different platforms and different encodings. **cmap2enc** only reads the first table that implements Unicode, that is, the first table with either platformID = 0 and encodingID = 3 or 4 or platformID = 3 and encodingID = 1 or 10. At least one of those will probably always be present, and (afaik) it doesn't matter which of these is used.

  The fact that only tableFormat = 4 is currently supported, however, might be more of a restriction, though I haven't yet found any font that used a different tableFormat.

- OpenType fonts implement features using so-called ‘Lookup Tables’. **cmap2enc** supports only features from the ‘GSUB’ table that have LookupType = 1 (single substitution).

  Some features are implemented as a combination of lookups from both the ‘GSUB’ and the ‘GPOS’ tables; these features are currently not supported. (An example are the ‘sinf’ (Scientific Inferiors) and ’subs’ (Subscripts) features of Linotype Palatino; these are implemented by first replacing the standard numerals with smaller ones and then adjusting their position.)

- Some of the font’s features may yield surprising results. E.g., Linotype Palatino doesn’t have small-caps forms of dotlessi, germandbls and the standard f-ligatures, so it will give you the lowercase forms of these glyphs even when the ‘smcp’ feature is specified. (In the *examples/palatinox* subdirectory, you’ll find a file *unsetSCaps.mtx* that deletes these lowercase forms; the *newlatin.mtx* file that’s part of *fontinst* then fakes small-caps forms of these glyphs.)

- If the user-defined substitution file substitutes glyph1 with glyph2, **cmap2enc** will also substitute glyph2 with glyph2; this may seem silly, but it means that any selected feature that would substitute glyph2 with glyph3 gets overridden.

- The glyph names are independent of the features selected, e.g., when using features like ‘smcp’ or ‘onum’, glyph names still come out as ‘a’ and ‘zero’ rather than ‘Asmall’ and ‘zerooldstyle’. When installing the font with *fontinst*, you should therefore always use *t1.etx* rather than *t1c.etx* or *t1j.etx*.

SEE ALSO

afm2afm, *autoinst*, *font2afm*, *ot2kpx*, *pfm2kpx*.

AUTHOR

Marc Penninga <marc@penninga.info>

HISTORY

2005−01−10 First version

2005−02−18 Rewrote some of the code, fixed a few bugs

2005−03−08 Input files searched via **kpsewhich** (where available)

2005−03−15 Input files searched using **kpsewhich** or **findtexmf**

2005−03−21 Fixed somed bugs; added ‘titl’ feature

2005−04−29 Improved the documentation

2005−08−01 Removed some warning messages that didn’t convey any useful information; added contents of *glyphlist.txt* to the program itself and removed the file from the *fontools* collection; expanded the example in the documentation