

NAME

ogonkify – international support for PostScript

SYNOPSIS

ogonkify [**-p** *procset*] [**-e** *encoding*] [**-r** *Old=New*] [**-a**] [**-c**] [**-h**] [**-t**] [**-A**] [**-C**] [**-H**] [**-T**] [**-AT**] [**-CT**] [**-ATH**] [**-CTH**] [**-E**] [**-N**] [**-M**] [**-mp**] [**-SO**] [**-AX**] [**-F**] [**-RS**] [**--**] *file* ...

DESCRIPTION

ogonkify does various munging of PostScript files related to printing in different languages. Its main use is to filter the output of Netscape, Mosaic and other programs in order to print in languages that don't use the standard Western-European encoding (ISO 8859-1).

SUMMARY USAGE

Installation instructions are provided in the file `INSTALL`. Assuming the installation has been correctly completed, save the PostScript output of Netscape or Mosaic to a file, say **output.ps**. Then print it using

```
% ogonkify -AT -N output.ps | lpr
```

in the case of Netscape, or

```
% ogonkify -AT -M output.ps | lpr
```

in the case of Mosaic.

You may want to change the **-AT** option to **-CT** in order to use a high quality Courier font from IBM (at the price of slower printing).

An alternative way to print from Netscape is to set the printing command in the printing dialog box to:

```
ogonkify -AT -N | lpr
```

For more details, see the `USAGE` section below.

OPTIONS

- p** Includes the specified procset in the output file.
- e** Set the encoding of the output. Defaults to **L2** (ISO 8859-2, a.k.a. ISO Latin-2). Other possible values are **L1** (ISO 8859-1, a.k.a. ISO Latin-1), **L3** (ISO 8859-3, a.k.a. ISO Latin-3), **L4** (ISO 8859-4, a.k.a. ISO Latin-4), **L5** (ISO 8859-9, a.k.a. ISO Latin-5), **L6** (ISO 8859-10, a.k.a. ISO Latin-6), **L7** (ISO 8859-13, a.k.a. ISO Latin-7), **L9** (ISO 8859-15, a.k.a. ISO Latin-9), **CP1250** (Microsoft Code Page 1250, a.k.a. CeP), **ibmpc** (Original IBM-PC encoding), **mac** (Apple Macintosh encoding) and **hp** (HP Roman Encoding).
- r** Use the font *New* in place of *Old*. Will lead to ugly or unreadable output when the metrics mismatch.
- a** Do the right font remappings for using Courier-Ogonki in place of Courier (the **a** stands for Adobe Courier). This avoids downloading any fonts to the printer.
- c** Do the right font remappings for using IBM Courier in place of Adobe Courier.
- t** Do the right font remappings for using Times-Roman-Ogonki in place of Times-Roman.
- h** Do the right font remappings for using Helvetica-Ogonki in place of Helvetica.
- A** Like **-a** but also downloads the Courier-Ogonki fonts.

- C** Like **-c**, but also downloads the IBM Courier fonts.
- H** Like **-h**, but also downloads the Helvetica-xxx-Ogonki fonts.
- T** Like **-t**, but also downloads the Times-xxx-Ogonki fonts.
- CT** Equivalent to **-C -T**.
- CTH** Equivalent to **-C -T -H**.
- E** Add the **Euro** currency sign to all standard fonts (use with **-e L9**).
- N** Do **Netscape** processing.
- M** Do **Mosaic** processing.
- mp** Do **mp** processing. Will not work with the **-A** option (use **-C** instead).
- SO** Do **StarOffice** processing.
- AX** Do **ApplixWare** processing.
- F** Do **XFig** processing.
- RS** Recode standard fonts. This is likely to work with applications that leave fonts in **AdobeStandardEncoding**, typically applications that do not even support printing even of characters.
- End options.

USAGE

Let us assume that you want to print a WWW page encoded in ISO Latin-2. Netscape stubbornly insists on printing it as ISO Latin-1. By using the File->Print command, have Netscape send the output to a file, say `alamakota.ps`.

As **ogonkify** is configured for ISO Latin-2 by default, passing it the PostScript generated by Netscape will correct the encoding of the fonts. It is enough to do:

```
% ogonkify -N <alamakota.ps | lpr
```

However, most printers do not have fonts with the needed characters installed; synthesized fonts will be downloaded and used instead of Courier and Times-Roman with **-AT**, and a very good Courier font from IBM will be used with: **-CT**. The command will therefore typically be:

```
% ogonkify -N -AT <alamakota.ps | lpr
```

or eventually

```
% ogonkify -N -CT <alamakota.ps | lpr
```

Typical usage with other programs is:

```
% ogonkify -M -AT <alamakota.ps | lpr
% ogonkify -mp -AT <alamakota.ps | lpr
% ogonkify -SO -AT <alamakota.ps | lpr
% ogonkify -AX -ATH <alamakota.ps | lpr
% ogonkify -XF -ATH <alamakota.ps | lpr
```

BUGS

Characters with an ‘ogonek’ should be constructed differently (for instance, the ‘ogonek’ used with an ‘a’ should be differently shaped than the one used with an ‘e’.)

It would be better to patch the programs we have the sources to than to post-process the produced PostScript.

The program is written in Perl.

NOTES

In order to view the output PostScript with Ghostscript, you might need to run **gs** with the flag **-dNOPLATFONTS**, and **ghostview** with the flag **-arguments -dNOPLATFONTS**.

Netscape, IBM, Adobe, PostScript, StarOffice, ApplixWare and possibly others are registered trademarks.

THANKS

Much of the composite character data have been provided by Primoz Peterlin, H. Turgut Uyar, Ricardas Cepas, Kristof Petrovay and Jan Prikryl.

Jacek Pliszka provided the support for **StarOffice**. Andrzej Baginski provided the support for **ApplixWare**.

Markku Rossi wrote **genscript** and provided many useful encoding vectors with the distribution.

Throughout writing the Postscript code, I used the **ghostscript** interpreter, by Peter Deutsch.

Larry Wall wrote **perl**, the syntax and semantics of which are a never ending source of puzzlement.

AUTHOR

Juliusz Chroboczek <jec@dcs.ed.ac.uk>, with help from loads of people.