

**Generate banners with pkbanner**

# Table of Contents

<b><u>Chapter 1. Generate banners with pkbanner</u></b> .....	<b>1</b>
<u>Reference</u> .....	1

# Chapter 1. Generate banners with pkbanner

Last modified on \$Date: 2005-10-10 00:04:12 +0200 (lun 10 de oct de 2005) \$

CUPS integrated banner facility sucks because banners can only be static PostScript files, even if they can include a few dynamic datas. Also it is possible to have several banners and jobs mixed in an unexpected way because of the way CUPS works.

To fix this problem for good, PyKota can use its own banners, which can be :

- Any static file, not only PostScript ones.
- The output of any executable command of your choice, launched at printing time, and to which are made available a bunch of environment variables which can help your command create a purely dynamic banner on the fly. In particular your command knows all about the status of the current user's print quota information.

Although you can use any command to generate banners for you, you can also use the **pkbanner** command line tool which is included in PyKota. It was designed for ease of use, without any compromise on its flexibility.

For example **pkbanner** allows you to specify a particular logo and url to be printed on the banner pages, as well as define a transparency factor to save toner while printing banners. You can also choose a particular page size to use, as well as include additionnal textual information on the banner page.

**pkbanner** generates PostScript code, but PyKota accepts that you pipe its output into any other command if you ever need to use another file format for your printer.

**pkbanner** is not meant to be launched from the command line though, but only through the `startingbanner` and `endingbanner` directives in `pykota.conf`.

---

## Reference

**pkbanner** [-v | --version] [-h | --help] [-l *image* | --logo *image*] [-p *size* | --pagesize *size*] [-s *luminosity* | --savetoner *luminosity*] [-u *link* | --url *link*] [*additionnal info*]