5.2 `xpick`—Pick images on an X11-screen

Evgueni Tcherniaev MT

The program **xpick** lets you pick an image from an arbitrary window or rectangular area of an X11-server and write it to a file in a variety of formats, where the latter is defined by the extension used for the file name.

```
xpick [-] [<file>]
```

xpick accepts only one option (-), which, when specified, picks an image from the window under the mouse pointer. This option is intented to pick images from pop-up menus, which are on the screen only when a mouse button is pressed and disappear immediately after the button is released.

List of possible file extentions

- `ps` A PostScript file with a compressed image. The image is centered, rotated and scaled to fill the maximum space on a page. This format is convenient for the preparation of transparencies and posters.
- `eps` An Encapsulated PostScript file with a compressed image, keeping the original dimension and containing a BoundingBox starting at `(0,0)`. It is intended for insertion into other documents.
- `epsi` Similar to the **eps** format, but a black and white device-independent preview image is added to the file. This can be used with text processing packages (e.g., Framemaker) to see an approximation of the image on the screen.
- `gif` Graphics Interchange Format (GIF). Use this format when you want to keep files or transfer them to other computers. It is also convenient for visualisation on screen, e.g., with the **xv** program of John Bradley (used by Mosaic).
- `pcx` The PCX format is used by IBM PC.
- `pict` The PICT format is used on Macintosh. It is intended for transferring images to a Macintosh, although the GIF format can also be used (e.g. there is a Macintosh GIF Converter by Kevin A. Mitchell).
- `ppm` The Portable PixMap format, which is part of the PBM+ library by Jef Poskanzer. Use it and a routine from the PBM+ library if you wish to have the image in a format not mentioned above.

The main differences between **xpick** and similar tools for capturing screen information of an X Windows server, like **xgrabsc**, **xwd**, **xv**, and **XtoPS** are:

- the manner in which the user selects the rectangular area to be picked: the blinking rectangle surrounding the area to be picked and involment of the keyboard input in addition to the mouse help to get rid of numerous parameters, nevertheless satisfying in quite a natural way the most common needs of users, namely, picking an image from a selected window or from an arbitrary rectangular area and a possibility to change the selected image just before outputing to a file.
- the compression scheme used for image encoding in Postscript files: **xpick** uses the Lepmpel-Ziv Welch (LZW) compression algorithms, thus producing very compact files (3-5 times less than files produced with Run Length Encoding (RLE) and 10-30 times less than files produced without compression)

How `xpick` works

The simplest way to invoke **xpick** is just to type

```
xpick
```

You will then be prompted to supply an output file name.

Alternatively, you can also specify the file name directly on the command line, e.g.,

```
xpick examp.gif
```

When **xpick** is invoked, the user sees a blinking rectangle surrounding the contents of the window in which the mouse pointer is currently placed. When the mouse is moved to a different window, the perimeter automatically changes to the size of the new window. If the mouse pointer is placed on the Window Manager border of a window, then the blinking rectangle will surround the window together with the Window Manager border. To select the image inside the blinking rectangle it is sufficient to click the left mouse button.

If a user-defined perimeter is required, then hold down the left mouse button to choose the first corner of the perimeter and then drag the mouse to define the opposite corner. The blinking rectangle will expand with the movement of the mouse.

After the rectangle has been selected the user should press the `<SPACE>` bar to pick the image. Before pressing the `<SPACE>` bar the user may to use the mouse to produce, if required, some changes inside the selected area, for example, to change the palette or display a pop-up menu.

More about PostScript files

PostScript files produced by **xpick** automatically understand on what kind of printer they are being output. On level 2 PostScript printers the native LZW decoding filter will be invoked, while on level 1 printers the inline decompression code is executed. As seen in Table 5.1 on the following page, this inline code is even more efficient than RLE decompression codes, as they are implemented in most other PostScript image compression tools.

Also, for files generated with the **.ps** extension, the image adapts itself to the dimensions of the output medium used on the printer, e.g., A4, A3, or A1. and, if necessary, the image will be rotated to fill the maximum amount of space on the page.

Benchmarks

In Figure 5.4 you see an image that was used for testing PostScript pixmap rendering times on various printers.

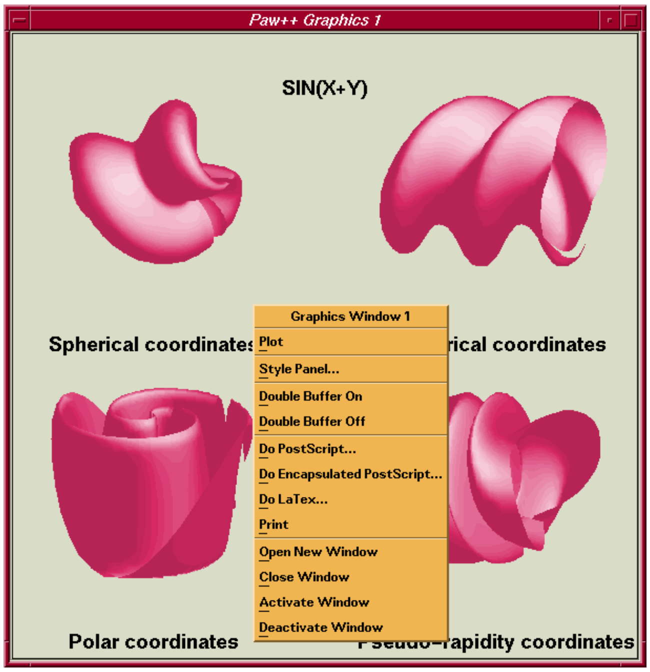

Figure 5.4: An example of a screen dump generated with `xpick` from a PAW++ graphics window

It is a colour image with a size 612x631 pixels and corresponds to a PAW++ graphics window. The original image was grabbed on a computer workstation screen by `xpick` and transformed into a `.gif` file. Then it was visualized using the `xv` program and saved into three different PostScript files, one containing an uncompressed image, one with a RLE compressed image and one with a LZW compressed image. The first two files were obtained using the save facility of the `xv` program, while the last one was made using `xpick` itself. Then a small PostScript routine was added to the three files to measure the rendering time and the files were printed on five different printers. The results are presented in table 5.1. As you can see both compression methods (RLE and LZW) save a lot of disk space, but files produced by `xpick` also print much faster, especially on level 2 PostScript printers.

`xpick` and `dvips`

Some old PostScript spoolers or post-processing programs get confused by comment statements inside Encapsulated PostScript files. Therefore the **`dvips`** program, by default, eliminates PostScript comment lines in EPS files, i.e. those starting with a `%` in column 1, from the generated output file. As `xpick` uses the PostScript level 2 compression algorithm and the latter sometimes generates lines starting with that character, you must use **`dvips`** with the option `-K0`, e.g.,

```
dvips -oout.ps -K0 infile.dvi
```

How to obtain `xpick`

At CERN the `xpick` program is available on all UNIX platforms, on VMS, and on Alpha. The files needed to compile the `xpick` program are available on the file server `asis01.cern.ch` in the directory `/pub/cern`. For possible problems the author can be contacted via e-mail at `chernaev@vxcern.cern.ch`.

612x631 colour image			PostScript level 2 printers			PostScript level 1 printers	
Program	Compression Scheme	size (kbytes)	Tektronix Phaser PXi colour 513-cp2	LaserWriter PRO630 grayscale 513-pub	QMS 1725 grayscale 513-qms	Tektronix Phaser PX colour 513-cp1	LaserWriter II grayscale
`xv`	uncompressed	2,352	3 min 01 s	7 min 57 s	1 min 17 s	5 min 18 s	13min 45 s
`xv`	RLE	146	3 min 27 s	6 min 29 s	1 min 52 s	9 min 53 s	19min 32 s
`xpick`	LZW	47	1 min 17 s	55 s	53 s	7 min 38 s	3min 20 s

Table 5.1: Rendering times on various printers