Xap

Button Bar

www.photocase.de

A program launcher doesn't always have to be a menu or a desktop icon. Xap takes an unconventional approach to this task and thus makes users rethink.

BY ANDREA MÜLLER

Panels, desktop icons, or menus are the traditional approaches to launching programs on a GUI-based desktop. However, none of these solutions made the developer of *Xap* (X application panel), a.k.a Rasca, really happy. He didn't feel like navigating menu hierarchies, or filling his desktop with icons. His Xap [1] utility takes a different approach. It organizes buttons for launching his favorite programs in windows on the desktop to save space – you can even use tabs. The tool comes complete with its own file manager.

Mix and Build

You will need to compile the source code for the tool. The usual three card trick of *./configure ; make ; su -c make install* won't get you far this time. The *configure script*, which is supposed to discover the location of the developer files, returns incorrect results, and adds the wrong paths for the **gtk** and *glib* **header** files to the makefile. To prevent this, you need to tell the *configure* script exactly where the include files are. Users of Suse Linux 9.0 can type the following to do so:

```
CFLAGS="-I/opt/gnome/include➘
/gtk-1.2 -I/opt/gnome➘
/include/glib-1.2" ./configure
```

Redhat 9.0, Suse 8.2, and Mandrake Linux 9.0, 9.1, 9.2, and 10.0 users will find the developer files for *gtk* and *glib* below */usr*. You need to enter:

```
CFLAGS="-I/usr/include/gtk-1.2 ➘
-I/usr/include/glib-1.2" ➘
./configure
```

-I means that the directory that follows, for example, */opt/gnome/include/gtk-1.2*, has the Include files needed by the compiler. The syntax above adds the paths to the *makefile*. You can then go on to run *make*, which should execute without any errors now. As an administrative user, type *make install* to copy the binary executable, *xap*, to */usr/local*. You can ignore the error message at the end. Xap includes links for the KDE 2 menu and is unable to copy them.

Desktopia

Only you can decide how your desktop looks. With deskTOPia we regularly take you with us on a journey into the land of window managers and desktop environments, presenting the useful and the colorful viewers and pretty toys.

Nicely Set Up

The following command

```
xap &
```

launches the new tool. The first time you start the program, it will create its own configuration directory, *~/.xap*, and populate the directory with links to its own tools. The program launches in a window that you can move around your desktop as required. The window has two tabs called *apps* and *tools*. *apps* is populated with quick start icons for the programs included with the Xap distribution: the *xwf* file manager, the *xfi* search tool, a front-end for GnuPG, and privilege management tool (see Figure 1).

The *tools* tab will be empty at this stage. Xap checks its config directory, *~/.xap*, to discover the applications it is expected to manage. Each folder below that directory represents a Xap tab. The contents of the *~/.xap/tools* directory are displayed as programs in a tab called *tools*. To add a program, use the *ln* command to create a link to the executable within a directory. For *xterm* we need:

GLOSSARY

gtk: *Short for the Gimp Toolkit, a graphics library that contains elements required to draw menus and dialog boxes. Many applications, including the latest version of Gimp, already use gtk-2.*

Header: *These are files that typically have a .h suffix and contain library interface definitions. If a program needs library functions, you need to include the header files for the library. Header files are normally located in packages with the same stem name as the library plus a dev or devel suffix.*

```
ln -s /usr/bin/xterm ~/.xap➥
/tools/xterm
```

You need to relaunch Xap to tell the tool to apply the changes.

You can add new tabs by selecting *New Page* in the *xap* pull-down menu (see Figure 2). This tells Xap to create a new folder below *~/.xap* where you can add links to programs.

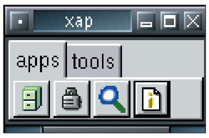


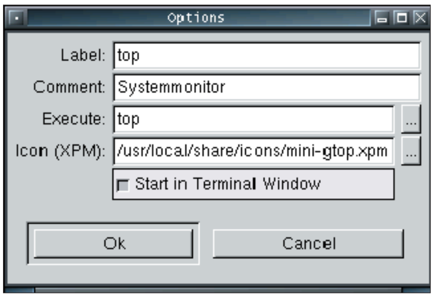
Figure 1: Xap automatically creates icons for its own collection of tools.

Pretty Desktop

Xap creates an icon for the *xterm* entry we just created, as it knows the application. It is not so friendly to unknown apps, however. They have to make do with a sad looking gray box. To brighten up the buttons, simply copy an XPM formatted graphic to the *~/.xap/.icons* directory. To tell the program launcher which button the graphic belongs to, add a name using the following conventions: *mini-name_of_link.xpm*. For example, if you have a link for *sylpheed*, *mini-sylpheed.xpm* would be the right name for the image. There is an even easier way of adding an icon to the launcher. Right-click the launcher and select *Options* to pop up the configuration dialog (see Figure 2). Enter the path to the icon to *Icon (XPM)* – with some luck you might find a suitable icon in the collection of some 100 that *xap* copies to */usr/local/share/icons* during the installation phase.

Fine Tuning

The pull-down menu for each tab has a few more features. Select *Rename Page* to rename a tab, or pop up a quick launch window by selecting *Execute*. The *Start application in a terminal* item is useful for applications that need a console, such as the system monitor, *top*.



Label: top
Comment: Systemmonitor
Execute: top
Icon (XPM): /usr/local/share/icons/mini-gtop.xpm
□ Start in Terminal Window

Figure 2: Use the button options to select an icon and define the executable.

How do you go about defining an icon to launch *top*? You won't get far with a link, as *top* needs to run in a terminal window as a "container". Luckily, the developer put some thought into this. The following command

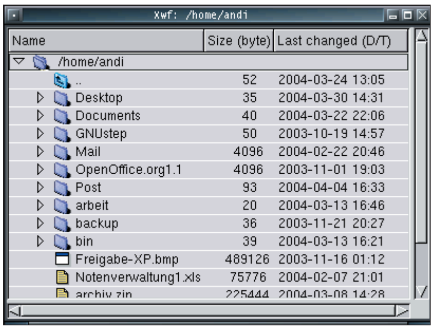
```
ln -s /usr/bin/top ➥
~/.xap/tools/top
```

creates a link and pops up a configuration dialog. In the *Options* for the link, select *Start in Terminal Window* and save your changes by clicking on *OK*. Under the hood, *xap* converts the link to a definition file. The file includes details of the icon that *xap* will display, the tooltip text to display on mouse over, and the command that *xap* executes when a user clicks the icon.

File Organizer

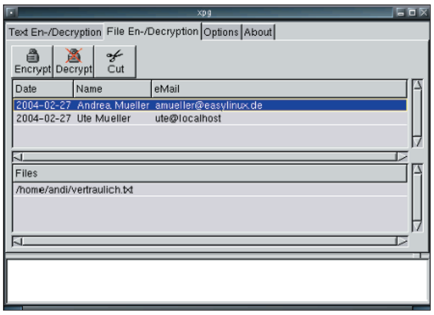
After creating your quick start icons, it's time to investigate the *xap* tools in the *apps* tab. The most important tool is the one hiding behind the button on the left, the *xwf* file manager (see Figure 3). The file manager allows access the other three programs, *xfi*, a *find* front-end, *xat*, for displaying and editing file permissions, and *xpg*, a GnuPG front-end.

As the file manager has a single window view, it makes sense to launch two instances. This allows you to copy and move files using drag & drop. Xwf will copy by default. If you use the center button on your mouse for the file copy, the file manager pops up a menu when you release the button in the target directory. You can then decide whether you want to *Copy* or *Move* the files, or create a *Link*. There is also *Copy (preserve)* that allows you to keep the permissions and time stamp for the file while copying.



Name	Size (byte)	Last changed (D/T)
/home/andi		
..	52	2004-03-24 13:05
Desktop	35	2004-03-30 14:31
Documents	40	2004-03-22 22:06
GNUstep	50	2003-10-19 14:57
Mail	4096	2004-02-22 20:46
OpenOffice.org1.1	4096	2003-11-01 19:03
Post	93	2004-04-04 16:33
arbeit	20	2004-03-13 16:46
backup	36	2003-11-21 20:27
bin	39	2004-03-13 16:21
Freigabe-XP.bmp	489126	2003-11-16 01:12
Notenverwaltung1.xls	75776	2004-02-07 21:01
archiv.zip	225444	2004-03-08 14:28

Figure 3: Xwf provides file management facilities for users.



Date	Name	eMail
2004-02-27	Andrea Mueller	amueller@easylinux.de
2004-02-27	Ute Mueller	ute@localhost

Files
/home/andi/vertraulich.txt

Figure 4: Xpg is graphical front-end for the gpg encryption tool.

Double-clicking a file pops up a dialog where you can associate a program with the file. The file manager will launch executables. If you need to edit a script, avoid double-clicking and select *Open with* in the context menu.

To create a permanent file association between PNG images and Gimp, select *Register* in the context menu for a PNG file. In the dialog box type *gimp* to automatically launch the image manipulation program whenever you double-click a file with the *.png* extension.

Search and Encrypt

You can launch a search operation by first selecting a folder and then *Find* in the context menu. Enter the search *Pattern*, and in the *Type* check boxes, specify what you want to find:*Plain file*, *Directory*, or *All*. Click on *find* to launch the search, and display the matches in the lower part of the window. If you want to avoid using the context menu, you can simply drag & drop the folder onto the *xfi* icon (the magnifying glass) in the Xap window.

To encrypt a file, either select *Encrypt* in the context menu, or or drop it onto the *xpg* icon. The program displays all the public keys in your keyring in the top panel (see Figure 4). Select the key for the user you want to send an encrypted file. Click on *Encrypt* to finish off the job.

Xap's drag & drop capabilities and the tools included with the distribution make a powerful team. Xap is extremely flexible and gives users the opportunity to take a customized environment with them no matter what kind of machine they are working on. ■

INFO

[1] Xap: *http://mpx.freeshell.net/*