deskTOPia: wmdrawer

Top Drawer

Program launchers are two a penny. To avoid being just one in a crowd, a launcher really needs to have some special features. If you're looking for a launcher with something special, try *wmdrawer*. **BY ANDREA MÜLLER**

An odd menu, a task bar here and there, and dozens of icons. Some people's desktops look like a bomb just hit them. Sound familiar? If you're tired of looking for lost applications, it may be time to clean up. *wmdrawer* [1] gives you a useful piece of desktop furniture that organizes icons you rarely need without sacrificing ease of access. As the name suggests, *wmdrawer* is a Dock Applet for Window-Maker [2].

wmdrawer is a retractable, animated drawer that serves as a button bar for your application icons. Most people don't even need to change their window managers to use *wmdrawer*. Because of their popularity, dock applets are supported by window managers like Blackbox, Fluxbox [3], PWM2, Waimea, and its successor Kahakai. Even KDE has a *Dock Application Bar* panel for dock applets.

Do-It-Yourself

Because most distributions do not include *wmdrawer*, you will

Figure 1: "wmdrawer" with drawers opening top down.

typically need to build the program launcher from the source code. This is not like buying things down at the furniture store, so you can put away that screwdriver, but you will need the *gtk*, *glib*, *gdk-pixbuf*, and corresponding developer libraries, which are identifiable by their *-devel* or *-dev* tags.

After fulfilling all these conditions, run *make* to compile *wmdrawer*, then *su* to root and call *make install* to copy *wmdrawer* to a directory below */usr/local*.

Deployment

If you type

```
wmdrawer &
```

the first time you launch *wmdrawer*, don't be surprised when the program refuses to run. The problem may be that *wmdrawer* is trying to read the configuration file *.wmdrawerrc*, before you have created the localized version of the file that will reside in your home directory. To avoid having to start from scratch, the program's author, Valéry Febvre, provides a sample configuration. The *make install* step copies the *wmdrawerrc.example* file to */usr/local/share/doc/wmdrawer*. To edit the file, copy it to the your home directory, and rename it .*wmdrawerrc*.

The next item on the list is modifying the configuration in your favorite editor. The sample file provides details on every option and looks a lot like a Windows *ini* file. An entry in square

brackets introduces a new section, and each line that follows has an option for the section. A pound sign (#) at the beginning of the line tags the line as a comment, telling *wmdrawer* to ignore the content.

As the name would suggest, the *[general]* section is for general settings. The line with

Figure 2: Program launcher with two "[column]" sections.

```
direction number_from_0-3
```

tells *wmdrawer* in which direction to open its virtual drawers. Your selection should reflect the position of the dock on your desktop. If the dock is at the top of the screen, *0* is the right selection, as it opens the drawers downward (Figure 1). The other options are *1* (from right to left), *2* (bottom up) and *3* (from left to right).

If you like a transparent look, try the

```
transparency 1
```

option in the *[general]* section. This gives the program icon background a transparent appearance. If it finds this line in *.wmdrawerrc*, the program will ignore an

```
icons_bg    image.xpm
```

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.

entry defined as a background graphic. If you use **pseudo-transparency**, *highlight 2* is a good idea. It tells *wmdrawer* to change the icon color on mouse over, where the line with the *highlight_tint* parameter defines the color. You need to add the color value in HTML notation, for example, *#0000FB* for an intensive blue. The KDE tool *kcolorchooser* tells you the values for the color.

To allow desktop drawers to open and close automatically on mouse over, set *show_on_hover* and *hide_on_out* to *1*. The number that follows *animation_ speed* defines how quickly the drawer opens: the higher the number, the quicker the drawer.

The second section in *.wmdrawerrc* is *[images_paths]*; it defines the thumbnail folder. The defaults are */usr/share/ pixmaps* and */usr/local/share/pixmaps*. You can add directories to the list; one folder per line. *wmdrawer* can handle both PNG and XPM format. If you intend to use the *Crystal* icons provided by your KDE installation, add the */usr/share/ icons/crystalsvg/32x32/apps/* path here. On SuSE Linux, you will need to replace */usr* with */opt/kde3*.

Setting Up

The next sections in the *.wmdrawerrc* file define the program launchers. Each one of the launchers starts with *[column]*. Individual columns can store multiple launchers, which *wmdrawer* organizes in multiple rows. The launcher for a single column is defined in a single row of the file.

The sections look like this

```
(tooltip) (image) (command)
```

The sample configuration file has a few examples of program launcher defini- tions. The *tooltip* field stores the text that *wmdrawer* displays when you hover the mouse over an icon. The *image* field expects the name of the thumbnail, and the *command* field expects the command

Figure 3: Three "wmdrawer" instances on a single desktop.

that *wmdrawer* will execute when you click the icon. The following entry would launch KMail:

```
(KDE-Mailer) (kmail.png) (kmail)
```

assuming that *kmail.png* is stored in one of the folders defined in *[images_paths]*. *wmdrawer* looks better organized if you limit the tool to two *[column]* sections (Figure 2).

Up, Running, and Growing

After adding your first few apps to *wmdrawerrc*, you can really start to use the dock applet. You can click to open drawers and hover the mouse over an icon to tell *wmdrawer* to display the tooltips you defined previously. Left clicking launches the program. To launch multiple programs use the center or right mouse button instead: this tells *wmdrawer* to leave the drawers open.

If you are unhappy with a setting or need to add another launcher, there is no need to quit and relaunch *wmdrawer*. Simply edit *.wmdrawerrc* and save your changes. Every time you open a drawer, *wmdrawer* checks the resource file and adds the changes on the fly.

Too many icons will tend to clutter even the biggest desktop drawer. To group programs by topic, you might like to launch multiple *wmdrawer* instances,

for example, one for Internet and office applications. Each applet needs its own configuration file of course. You can launch the first instance of the program in the normal way (*wmdrawer &*) to tell it to parse the *.wmdrawerrc* file. When you launch the second instance of *wmdrawer*, you need to add the *-c* parameter, followed by the path to the configuration file, for example *wmdrawer -c ~/drawer2*. To distinguish between the two *wmdrawer* instances at a glance, use

```
dock_icon   graphic.xpm
```

to give the tool a different background (Figure 3). You need to add this state- ment to the *[general]* section of the configuration file used by the *wmdrawer* instance. The only danger then is that your desktop might start to look as untidy as it was before you installed *wmdrawer*, because of all of those draw- ers. (amü) ∎

INFO

[1] wmdrawer: *http://people.easter-eggs.org/ ~valos/wmdrawer/*

[2] *http://www.windowmaker.org*

[3] Andrea Müller: "Small Footprint, Quick, Practical," Linux Magazine, #43 / June 2004, p. 80.