An up-to-date look at free software and its makers

PROJECTS ON THE MOVE

Now that we have entered the age of three-dimensional desktops, suggestions on redesigning controls keep springing up like mushrooms. When Gnome 3.0 sees the light of day, will the menus be circular and include files alongside applications? **BY CARSTEN SCHNOBER**

Until a couple of years ago, the 3D desktop was considered to be a novelty that got in the way of meaningful work. Despite early skepticism, the major Linux distributions do not want to lose ground to Mac OS and Windows Vista, and all of them now offer user-friendly three-dimensional desktop support.

Wobbling windows, transparency, rotating cubes when you switch workspaces – these visual effects raise some doubt as to the 3D desktop's ability to help users work more efficiently. But at least they give the GUI a more modern and advanced look. Today's other major operating systems – Windows Vista and Mac OS X – have a similar look, and once you have grown accustomed to effect-packed window managers like Compiz Fusion [1], you will not want to do without them. Although the benefits might not be tangible, they do much to boost the user's feel-good factor.

If you need more evidence that the wobbling desktop on Linux has grown up over the past two years, just consider the fact that unnerving setup marathons – involving much trial and error and regular crashes – are now a thing of the past. Still, there is room for improvement. Although the KDE and Gnome desktops now integrate Compiz, users are still waiting for a paradigm shift in basic controls. So far, the problem of how to reconcile two-dimensional input devices, such as the keyboard and the mouse, with a 3D desktop is one that not many people have considered. Little has happened to change the WIMP model (windows, icons, menus, pointers) over the years.

Gnome, Gimmie, and GdMap

Cole Ansley suggested redefining the menus in Gnome. In contrast to the legacy approach, the circular application menu [2] no longer hides in a corner of the desktop to pop up when needed. Instead, the menu sits at the center of the desktop as a kind of wheel that spreads over any active windows. The Gnome main menu icons are spread around the circle. Clicking one of these icons pops up another wheel that displays the content of the corresponding sub-menu.

The Gnome circular application menu is designed for the 3D desktop and requires a compositing window manager that supports transparency. If you do not have a compositing window manager, the program will launch, but it will use a black background that completely covers the underlying desktop, which prevents any meaningful use.

Ansley had the idea for the new menu design last March, but he didn't start implementing it until December. The source code still does not offer convenient automake support. If you are interested in manually running *make* to build the program, you can download the source code from Subversion. To find the address, check out the *Source* tab on the project homepage [2], which is hosted on Google's Code subsection.

You can try the software without too many bugs spoiling your fun. If you intend to use the circular application menu on a regular basis, you might want to assign a keyboard shortcut or add an icon to the panel. Otherwise, you will need to launch the menu from the command line, which doesn't make much sense in the context of a graphical desktop.

Gnome developers tend to use Gnome

3.0 [3] as a screen on which to project new ideas, such as the circular desktop, because the version change, which has been in the cards for years, is still just an idea. Because Gnome developers do not see any need to redevelop major parts of the desktop to make the shift, there are no tangible plans for version 3 of the desktop environment.

The official Gnome 3.0 page offers a scratchpad [4], on which developers can present their ideas to the general public. No code backs up most of the proposals; instead, most developers simply want to trigger a discussion and present their own personal parallel universe with a view of the Gnome desktop's future.

Gimmie [5], another suggestion for developing the Gnome desktop, is a fairly tangible and working draft intended for the future Gnome 3.0. Gimmie is made up of two elements: a floating menu that resides at the bottom of the screen and only shows active programs, serving as a window changer and reminding users of the Apple Mac OS X menu, and a component that resides in a panel and provides four tabs, giving users access to programs and data. Windows can be opened as required via the Gimmie menu component.

The changes mean that applications no longer hog all the limelight. Instead, users can access a *Documents* tab with individual files sorted by categories such as *Music & Movies*. If you click one of the icons, the desktop environment finds a matching application and launches it to open the file. The *Contacts* tab follows the same principle and is where users can access contact addresses for people they address directly via mail or chat.

Integrating voice over IP calls is just a small step.

The menu elements change to reflect the user's habits – larger icons represent frequently used applications, documents, and contacts, and make it easier for users to find their favorites. A *Recently Used* section offers rapid access to programs and files on which the user has been working, independent of the underlying programs and files.

Pre-built Gimmie binaries exist for some distributions. Although Release 0.2.8 does have its fair share of bugs that occasionally cause crashes or waits, plans exist for the 0.3 series; however, the code is not currently usable. For now, I would not recommend using Gimmie in a production environment.

An intelligent desktop environment makes the best use of space available on the desktop, but it is not much help if you start to run out of disk space. Experience shows that finite disk space is not the answer. Space wasting is something that can only be prevented by organized file management. But if you are in a hurry, it is easy to drop a downloaded DVD image into the wrong directory. Typically, appeals to save space on multi-user systems are ignored.

To locate disk hogs, you first need to identify the biggest files. Because most people can interpret graphical displays more easily than text output in the style of *du*, various programs visualize disk space usage.

A new version of GdMap [6] (Figure 1) has brought the tool back into the limelight.

GdMap represents files as rectangles whose dimensions reflect their disk space requirements. Files in subdirectories are shown as elements of larger rectangles. A single color shows users that they share a storage location, and a double click lets the user navigate the directory structure.
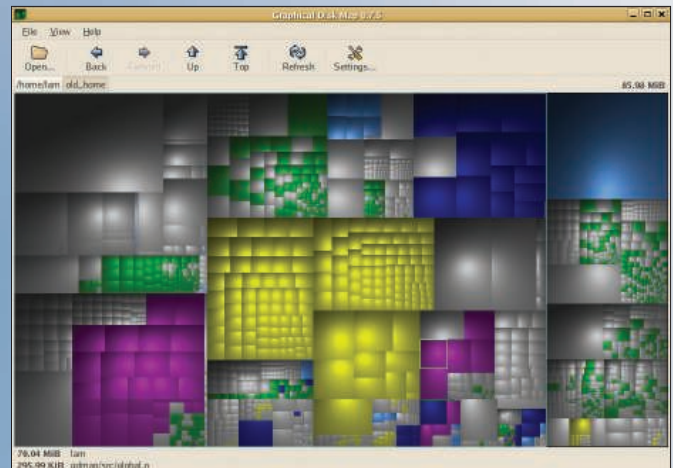

Figure 1: A new version of GdMap.

Besides fixing a couple of bugs, author Markus Lausser has implemented two new features in the new 0.8.0 version. For one thing, the software now detects filesystem limits when searching subdirectories and stops the search if configured to work this way. Sparse file handling is often a problem for analysis tools. Sparse files, which are typical for Unix, use far less disk space than their ostensible size suggests because they contain a large number of contiguous zero bytes, which do not need to be stored explicitly. Even in this special case, GdMap can now ascertain the correct file size.

The use of rectangles rather than circles – like those in Baobab [7], for example – provides no objective advantage; however, the subjective goal is to give users an overview of the scene at a glance. In contrast to the terse command-line output provided by *du*, disk hogs still catch the eye, even if you are in a hurry. ■

INFO

[1] Compiz Fusion: *http://www.compiz-fusion.org*

[2] Gnome circular application menu: *http://code.google.com/p/circular-application-menu*

[3] Gnome 3.0: *http://live.gnome.org/ThreePointZero*

[4] Gnome scratchpad: *http://live.gnome.org/ScratchPad*

[5] Gimmie: *http://beatniksoftware.com/gimmie*

[6] GdMap: *http://gdmap.sourceforge.net*

[7] Baobab: *http://live.gnome.org/GnomeUtils/Baobab*