Ion

Jumping Tasks

Many modern window managers are overloaded with features. Ion concentrates on the essentials and breaks new ground in doing so. This is a must for keyboard fans. Rather than burying your screen under a mountain of different window tasks, Ion cleverly tiles then onto the screen.

BY ANDREA MÜLLER

Is your desktop too cluttered or unmanageable with all of those overlapping windows? *Ion* provides a GUI comprising of multiple **Frames** tiled adjacently or vertically. Each frame can contain multiple application windows, so-called Clients. This takes the *Ion* window manager [2] a step further than *pwm* [1], which was also authored by Tuomo Valkonen, and also allows you to run multiple applications in a single window frame.

Installation – Special Request Only

Unfortunately, of all the major distributions only Debian provides Ion packages; this means working with a **compiler** on most other systems. The latest developer version of the Ion window manager proved so stable in our lab tests that we prefer to use it rather than the stable version, which is fairly ancient by now, dating from February 2002. There is a downside, however, as the developer version requires the *Lua* [3] programming language (see Box 1 for details on installing *Lua*). Apart from that, Ion is quite frugal in its use of resources, and requires on the *libltdl* library from the *libtool* package, version 1.4.3 or later.

After installing Lua, it's time to install Ion itself. To do so, unpack the source code by typing *tar -xzf ion-devel-20030627.tar.gz* and change to the new *ion-devel-20030627* subdirectory created by this step. However, before you start compiling the window manager, you will need to edit *system.mk* to suit your requirements. Listing 1 provides an example that shows you how to modify the installation path and disable **Xinerama** support. Typical alternatives to the defaults are available in the file; you just need to remove the hash sign (#) for the required options and add them to the appropriate standard lines.

You can now type **make depend** and **make** to compile Ion. Assuming that you are **root**, **make install** will then install your new window manager to /usr/local/.

Unfortunately, there is no patent recipe for launching Ion. As an example, you would configure the kdm login manager for your distribution using the KDE control center, adding a new type called Ion under **System Administration / Login manager / Sessions**.

If you do not use a GUI login manager, you will need to edit the file appropriate to your distribution – check the manual to find out which file this is. For most systems, an entry such as **exec /usr/local/bin/ion** in **~/.xinitrc** should do the trick, and allow you to launch the window manager of your choice when you enter the **startx** command.

By default. Ion uses run-mailcap as an editor and file viewer. Unfortunately, this program is not available for many distributions. To ensure that the view and edit function keys actually provide the expected functions, add the following (you will need **root** privileges):

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.

Box 1: Installing Lua

The Lua programming language is mainly used for configuring other programs. It is typically embedded in another program that will call Lua functions to perform specific tasks. Ion leverages this capability when a user calls a window action by pressing a keyboard shortcut. For example, [Alt-a] adds another window to the active frame. You can achieve the same effect by pressing [Alt-F3] to open a Lua command line and type *QueryLib.query_attachclient(_)*.

Lua does not come with a *configure* script, so you will need to edit the *config* file in the *lua-5.0* directory, after launching *tar -xzf lua-5.0.tar.gz* to unpack the tarball. The individual lines in the config file are well documented.

Now go on to compile and install Lua using the following commands:

```
make
make so
su
Enter your root password
make install
make soinstall
```

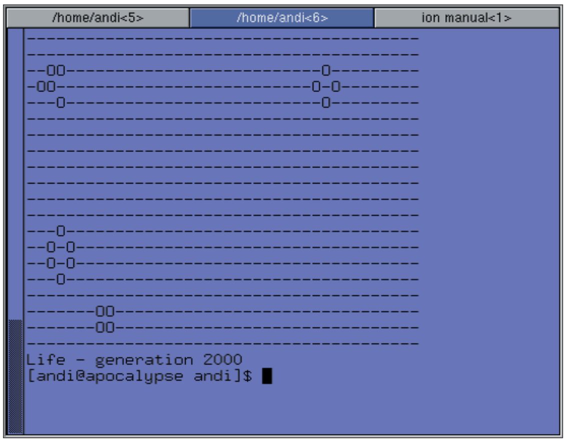
The *make so* and *make soinstall* are not typically required, but install the **shared libraries.**

```
#!/bin/sh
exec xterm -e vim "$1" > ➥
/dev/null
```

to the /usr/local/share/ion-devel/ion-edit file and the following to /usr/local/share/ion-devel/ion-view:

```
#!/bin/sh
exec xterm -e less "$1" > ➥
/dev/null
```

This example specifies vim as your editor and less as your viewer, but you can modify both commands as required.



```
----------------------------------------
----------------------------------------
--00--------------------------0---------
-00---------------------------0-0--------
---0--------------------------0---------
----------------------------------------
----------------------------------------
----------------------------------------
----------------------------------------
----------------------------------------
----------------------------------------
----------------------------------------
---0------------------------------------
--0-0-----------------------------------
--0-0-----------------------------------
---0------------------------------------
----------------------------------------
-------00-------------------------------
-------00-------------------------------
----------------------------------------
Life - generation 2000
[andi@apocalypse andi]$ █
```

Figure 1: Use the tabs to toggle between the programs in a frame

A Special Kind of Desktop

After all that installation work, you might be in for a disappointment when you launch Ion for the first time: an empty frame is all you get, and clicking will get you nowhere. Ion will support your mouse, but the easiest way of controlling the window manager is to use the keyboard (Table 1 contains a list of major functions). The commands followed by (CL) for command line, tell Ion to open a command line for further interaction.

To get started, you might like to press [Alt-F1] to open the Ion manpage that provides a description of Ion's navigational commands. [F2] opens an xterm in a new tab (see Figure 1). Now the frame has more than one tab, you can actually do something useful with your mouse, that is, click to switch tabs. Alternatively, use the keyboard shortcut [Alt-k] [n] to jump to the next window in the current frame.

Listing 1: system.mk

```
#Installation path
PREFIX=/usr/local

[...]

#Disable Xinerama support. If you
intend to run Ion on a multiple
screen System,
#leave the default
#XINERAMA_LIBS=-lXinerama
DEFINES += -DCF_NO_XINERAMA
```

Press [F3] to launch a program, such as the Mozilla Web browser for example. This opens a command line at the bottom of the screen, allowing you to type the required command. A [Tab] based auto-complete function just like in the Shell will save you a lot of typing (see Figure 2), and you can press [Ctrl-c] if you change your mind. Prepend a colon to launch text-based applications that run in a terminal window, for example:

```
:mutt
```

Effective Diversity

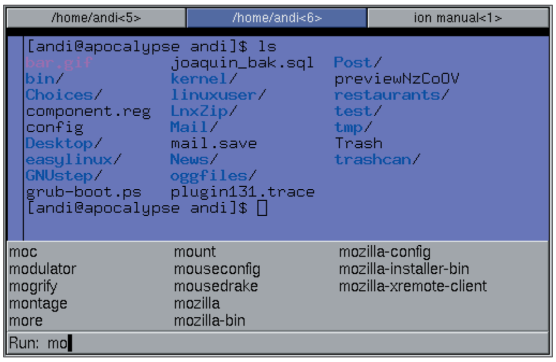
A single large frame may make sense for a browser, but there are situations where you need to monitor multiple programs. The keyboard shortcut [Alt-s] splits the current frame horizontally; to split vertically press [Alt-k] [s] instead. You can split frames as many times as you like, the only restriction being your screen size.

Pressing [Alt-r] and then the arrow keys allows you to scale tiled windows that you have created in this way; alternatively, click on the border of the window and, holding down the left mouse key, drag the window to the required size (see Figure 3). If you want to move an application to another frame, press down the center mouse key, drag the application tab to the target frame and drop the tab.

You can navigate the frame landscape by pressing [Alt-p] (up) and [Alt-n] (down), and press [Alt-Tab] to jump to adjacent frames. If things start to get cluttered, you can press [Alt-Enter] to expand a frame to full-screen display on a new desktop. Pressing [Alt-Enter] a second time reverts to the normal display and packs the program away in its original frame. [Alt-c] allows you to remove applications and superfluous frames.

My Desktop Collection

You are bound to run out of space sometime, no matter how big your screen is. In this case, press [F9] to open a command line that allows you to jump to an existing desktop or create a new one. Pressing [Tab] on the command line displays a list of the current desktops. If you specify a name that is not on the list, Ion will create a desktop with that name. You can also press [Alt-left arrow] and [Alt-right arrow] instead of [F9] to navigate the current desktops.



```
[andi@apocalypse andi]$ ls
bar.gif         joaquin_bak.sql  Post/
bin/            kernel/          previewNzCoOV
Choices/        linuxuser/       restaurants/
component.reg   LnxZip/          test/
config          Mail/            tmp/
Desktop/        mail.save        Trash
easylinux/      News/            trashcan/
GNUstep/        oggfiles/
grub-boot.ps    plugin131.trace
[andi@apocalypse andi]$ □
```

```
moc                 mount               mozilla-config
modulator           mouseconfig         mozilla-installer-bin
mogrify             mousedrake          mozilla-xremote-client
montage             mozilla
more                mozilla-bin
Run:  mo
```

Figure 2: The Ion command line has an auto-complete function

GLOSSARY

Compiler: *A program that translates source code to binary object code. This step is necessary before running a program written in a language like C or C++.*

Shared libraries*: Program libraries that have a similar function to Windows DLL files. Programs load the functions they need at runtime, thus saving hard disk space and memory.*

Xinerama: *A XFree86 extension that allows you to create a desktop that spans multiple screens.*

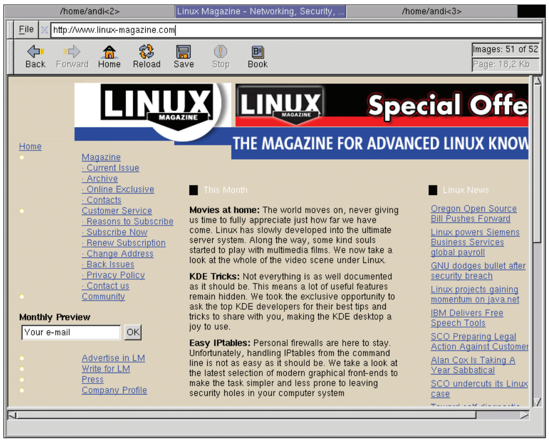
If you need an application from another desktop, there is no need to search for it: [Alt-a] opens a command line where you can use the auto-complete function to select and access the required window.

Although a frame based approach is practical, a desktop that allows you to move windows arbitrarily is better suited to some applications. The Gimp is one example of this. When working with The Gimp you typically want to have multiple windows on view simultaneously and at user-definable positions. The good news is that Ion supports so-called floating desktops with applications like The Gimp in mind. Floating desktops provide similar functionality to many other window managers. However, this mode is not accessible by shortcut in contrast to the tiled desktop. Instead, press [F9] and type the following in the command line

```
WFloatWS:NAME
```

where NAME is the name of the new desktop. Then press [F2] to launch xterm and call your programs. You can group windows in this mode by holding down the center mouse key and dragging the title bar of a window to another active application (see Figure 4).

You are in for a treat next time you start Ion, as the window manager saves and restores your desktops and frame arrangements without launching the programs that you were running when you quit the previous session.



Movies at home: The world moves on, never giving us time to fully appreciate just how far we have come. Linux has slowly developed into the ultimate server system. Along the way, some kind souls started to play with multimedia films. We now take a look at the whole of the video scene under Linux.

KDE Tricks: Not everything is as well documented as it should be. This means a lot of useful features remain hidden. We took the exclusive opportunity to ask the top KDE developers for their best tips and tricks to share with you, making the KDE desktop a joy to use.

Easy IPtables: Personal firewalls are here to stay. Unfortunately, handling IPtables from the command line is not as easy as it should be. We take a look at the latest selection of modern graphical front-ends to make the task simpler and less prone to leaving security holes in your computer system

Oregon Open Source Bill Pushes Forward

Linux powers Siemens Business Services global payroll

GNU dodges bullet after security breach

Linux projects gaining momentum on java.net

IBM Delivers Free Speech Tools

SCO Preparing Legal Action Against Customer

Alan Cox Is Taking A Year Sabbatical

SCO undercuts its Linux case

Figure 4: Two programs in a floating window

Settings

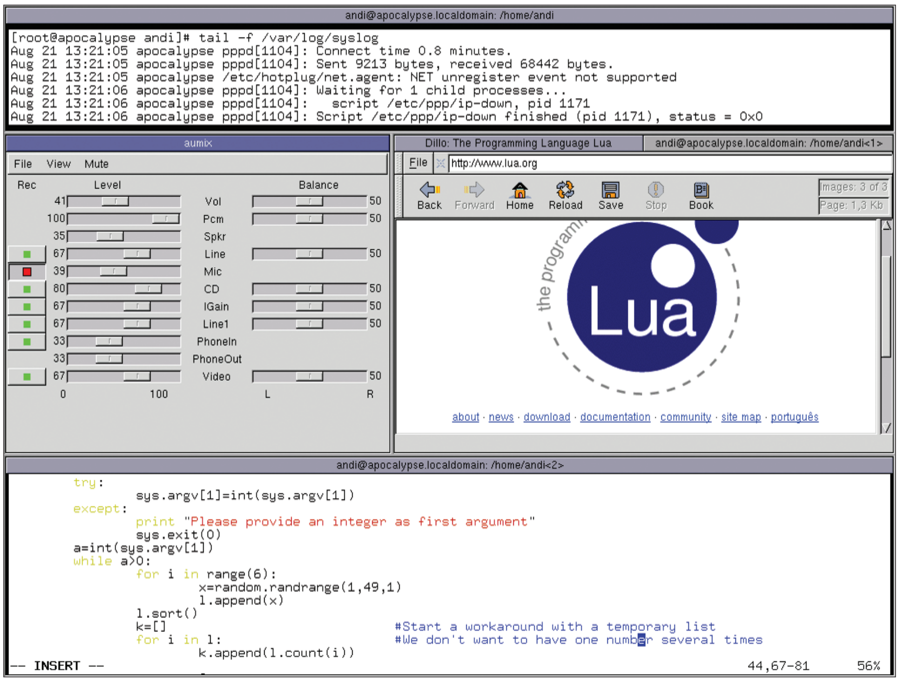
Ion's default configuration is stored in the /usr/local/etc/ion-devel/ directory. To modify the desktop to suit your own personal requirements, copy the ion-core.lua file from this directory to ~/.ion-devel/ and modify it to reflect your requirements. If you want Ion to display the frame contents while you are resizing a frame, for example, replace **FALSE** with **TRUE** in

```
enable_opaque_resize(FALSE)
```

You do not need an editor to change Ion's color scheme, however. Ion provides a number of new outfits as **look** files in the configuration directory. Simply create a **symbolic link** called **draw.lua** for the required look in the ~/.ion-devel/ directory. To imitate the style we used for our screenshots, enter the following

```
ln -sf /usr/local/etc/ion-devel↩
/look-greyviolet.lua ↩
~/.ion-devel/draw.lua
```

Ion should now sport a gray/violet color scheme when next launched. ■



```
[root@apocalypse andi]# tail -f /var/log/syslog
Aug 21 13:21:05 apocalypse pppd[1104]: Connect time 0.8 minutes.
Aug 21 13:21:05 apocalypse pppd[1104]: Sent 9213 bytes, received 68442 bytes.
Aug 21 13:21:05 apocalypse /etc/hotplug/net.agent: NET unregister event not supported
Aug 21 13:21:06 apocalypse pppd[1104]: Waiting for 1 child processes...
Aug 21 13:21:06 apocalypse pppd[1104]:   script /etc/ppp/ip-down, pid 1171
Aug 21 13:21:06 apocalypse pppd[1104]: Script /etc/ppp/ip-down finished (pid 1171), status = 0x0
```

```
        try:
                sys.argv[1]=int(sys.argv[1])
        except:
                print "Please provide an integer as first argument"
                sys.exit(0)
        a=int(sys.argv[1])
        while a>0:
                for i in range(6):
                        x=random.randrange(1,49,1)
                        l.append(x)
                l.sort()
                k=[]                                    #Start a workaround with a temporary list
                for i in l:                             #We don't want to have one number several times
                        k.append(l.count(i))
-- INSERT --                                                                         44,67-81      56%
```

Figure 3: Use the tiling functions to create variable-sized frames

Table 1: Major Shortcuts

Key	Function
[F1]	Display a manpage (CL)
[F2]	Launch xterm
[F3]	Launch a program (CL)
[F4]	Open an SSH connection (CL)
[F5]	Edit file (CL)
[F6]	View file (CL)
[F9]	Switch desktop or create a new desktop (CL)
[F12]	Quit Ion (CL)
[Alt-Arrow right/left]	One desktop right/left
[Alt-g]	Jump to another frame (CL)
[Alt-p/n/Tab]	One frame up/down/right
[Alt-c]	Close active object
[Alt-k] [c]	Kill active application
[Alt-k] [n/p]	Move to next/previous tab in a frame
[Alt-a]	Add application to active frame (CL)
[Alt-s]/[Alt-k] [s]	Tile frame horizontally/vertically

GLOSSARY

Symbolic link: *An entry in a filesystem that looks like a normal file or directory but actually points to a file or directory located at a different position in the filesystem.*

INFO

[1] Joachim Moskalewski: "Cleverly Contrived", Linux Magazine, Issue 18: March 2002, p77-79, *http://www.linux-magazine.com/issue/18/PWM.pdf*

[2] *http://modeemi.fi/~tuomov/ion/*

[3] *http://www.lua.org/*