

Issue #19 - November 2008



full circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU COMMUNITY



INTERVIEW :
EMILIO POZUELO MONFORT

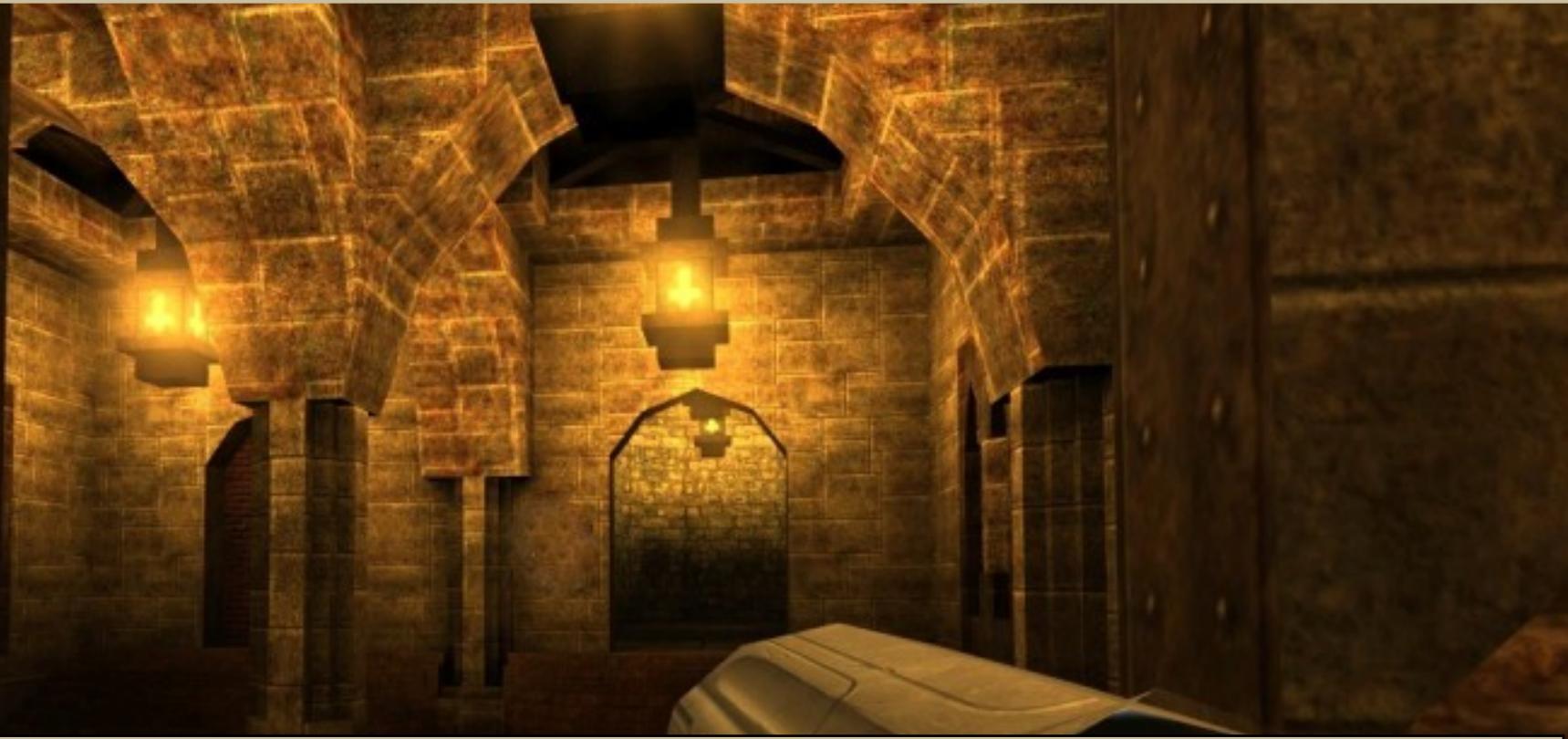
HOW TO :
GIMP - PART 8
PROGRAM IN C - PART 3
CREATE MOBILE MULTIMEDIA
CREATE A WIFI ACCESS POINT

COMMAND AND CONQUER :
LOST AND FOUND

NEW FEATURE :
UBUNTU GAMES

UBUNTU GAMES

REVIEW OF OPEN ARENA



Full Circle magazine is not affiliated with or endorsed by Canonical Ltd.



full circle

www.fullcirclemagazine.org



P.08



P.13



P.15



P.18



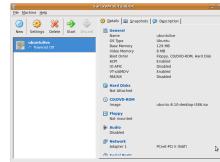
P.21



P.23



P.27



P.32



The articles contained in this magazine are released under the Creative Commons Attribution-Share Alike 3.0 Unported license. This means you can adapt, copy, distribute and transmit the articles but only under the following conditions: You must attribute the work to the original author in some way (at least a name, email or URL) and to this magazine by name ('full circle magazine') and the URL www.fullcirclemagazine.org (but not attribute the article(s) in any way that suggests that they endorse you or your use of the work). If you alter, transform, or build upon this work, you must distribute the resulting work under the same, similar or a compatible license.

Full Circle magazine is entirely independent of Canonical, the sponsor of the Ubuntu projects, and the views and opinions in the magazine should in no way be assumed to have Canonical endorsement.



News

p.04

Command and Conquer :

Lost & Found

p.06

How-To :

- Program In C - Part 3
- Create A WiFi Access Point
- GIMP - Part 8
- Create Mobile Multimedia

p.08

p.13

p.15

p.18

My Story :

Ubuntu John

p.20

Book Review:

Beginning Ubuntu Linux

p.21

Interview :

Emilio P. Monfort

p.23

Letters

Ubuntu Women

p.25

Ubuntu Games

p.27

Q&A

My Desktop

p.29

Top 5 : Virtualization Apps

p.32

How to Contribute

p.34

icons: KDE4 Oxygen



EDITORIAL



Welcome to another issue of Full Circle Magazine.

Not only does this issue have a new monthly section, Ubuntu Games, but it is the first issue with a competition! Apress Publishing kindly supplied us with a review copy of their book **Beginning Ubuntu Linux**, which is reviewed on [page 21](#), and have allowed us to offer the review copy as a prize (I've kept it in pristine condition, honest!). So have a read through the review; then email us your answer to the question at the end of the review. We hope to have a book review each month from now on.

Another first for **Full Circle** this month is that this issue is the first to be created in GNOME. That's right, I've switched sides. Why? The main reason being that most of the applications I used in KDE were GNOME based, so I was effectively running a GNOME/KDE hybrid to use Transmission, Firefox, Scribus, Thunderbird, et al. Moreover I wanted an excuse to upgrade to Intrepid. Scribus gave me some major headaches in KDE. I expect to be free from them under GNOME (he says, crossing his fingers!).

Anyway, enough babbling from me, enjoy the new issue, and good luck in the competition.

All the best,

Ronnie

Editor, Full Circle Magazine

ronnie@fullcirclemagazine.org

This magazine was created using :



What is Ubuntu?

Ubuntu is a complete operating system that is perfect for laptops, desktops and servers. Whether at home, school or work Ubuntu contains all the applications you'll ever need including word processor, email application and web browser.

Ubuntu is and always will be free of charge. You do not pay any licensing fees. You can download, use and share Ubuntu with your friends, family, school or business for absolutely nothing.

Once installed, your system is ready to use with a full set of productivity, internet, drawing and graphics applications, and games.

<http://url.fullcirclemagazine.org/7e8944>



NEWS

SCO loses Linux legal fight



The long-running saga of SCO versus everyone who ever so much as looked sideways at Linux – but especially Novell and IBM – has drawn to a close with the news that Federal District Judge Dale A. Kimball has dismissed all the company's claims.

According to Ars Technica's article on the judgement all claims against Novell have been dismissed, declaratory relief has been granted, and SCO is faced with a whopping bill for "unjust enrichment" of over \$2.54 million (£1.68 million) – plus interest.

Ending the five year attack on Linux and anyone who would seek to popularise it – and finally laying to rest claims that the popular open-source kernel and associated operating system somehow 'ripped off' SCO's UNIX intellectual properties – the judgement is likely to remain unpaid, as the company is currently undergoing bankruptcy proceedings in the face of a skydiving stock price.

Source: <http://www.bit-tech.net>

Linux Powers FTA Satellite Box



CaptiveWorks has announced a Linux IP set-top box (STB) and digital video broadcast receiver that offers FTA satellite HD video reception, and media center features. The CW-4000HD Linux Media Center is based on Gentoo Linux and other open source projects, says the company.

Based in Los Angeles, CaptiveWorks has designed, manufactured, and distributed Linux-ready digital video broadcasting receivers since 2005. Earlier products included the CW-1000S and CW-3000HD satellite STBs, as well as the earlier CW-600S, also aimed at FTA users.

FTA loosely describes a collection of digital video broadcast technologies that do not require subscription services to enjoy. Service availability varies greatly by country.

The CW-4000HD

The new CW-4000HD is a compact Linux desktop equipped with a satellite receiver and media playback capabilities. The core component is the FTA satellite receiver, which is designed to receive no-cost, unencrypted FTA satellite transmissions of HDTV and SDTV signals. Additional features include an electronic program guide, live-TV

pause, and zoom.

The CaptiveWorks system can also play FTA TV channels made available via IPTV broadcasts over the Internet, says the company. In addition, it supports peer-to-peer P2P-TV Internet video stream interfacing, in which users broadcast streaming video as well as receive it, somewhat like BitTorrent. An ATSC receiver for digital broadcast reception is optional, and is said to include cable TV support.

The CW-4000HD's media center capabilities include Internet radio reception, an MP3 player, and a picture viewer application. The box comes with a Firefox web browser, and offers DVR and the ability to burn videos to DVDRW, says CaptiveWorks.

Source: <http://www.linuxdevices.com>

FULL CIRCLE NEEDS YOU!

A magazine isn't a magazine without articles and Full Circle is no exception. We need your Opinions, Desktops and Stories. We also need Reviews (games, apps & hardware), How-To articles (on any K/X/Ubuntu subject) and any questions, or suggestions, you may have.

Send them to:
articles@fullcirclemagazine.org



NEWS

Amazon to Sell OLPC's XO Laptop



One Laptop Per Child confirmed it will start selling the XO laptop through Amazon.com on Monday.

The XO laptop will be sold through Amazon in the same way as the G1G1 program, where a consumer donates US\$400 for two laptops, with one of them delivered to a child in a developing nation.

Only Linux-based XO laptops will be available through Amazon, said Jim Gettys, vice president of software engineering at OLPC. A Windows version will not be sold.

For now, Amazon's U.S. storefront will sell the XO. OLPC is reviewing the possibility of selling the laptop through Amazon in other countries.

Designed for use by children in developing countries, the laptop has been praised for its innovative hardware features and environmentally friendly design. It comes with 1G byte of internal flash storage, 256M bytes of RAM, a 7.5-inch liquid-crystal display screen and wireless networking.

Source: <http://www.pcworld.com>

Canonical To Build ARM Version Of Ubuntu



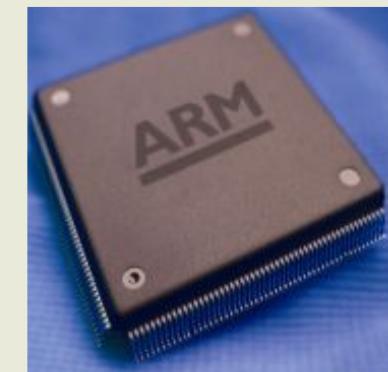
Canonical plans to build a version of Ubuntu to run on ARM's v7 processors, which are aimed at mini-notebooks and other mobile Internet devices.

The ARM version of the Linux-based desktop operating system is expected to be released in April. ARM competes with Intel's low-power Atom processor, which is gaining traction in so-called "netbooks," defined as sub-\$500 systems with screen sizes of 10 inches or less. The mini-notebooks, which typically run Linux or Windows XP, are primarily aimed at schoolchildren or at adults who want a light, compact system for checking e-mail and browsing the Web on the road.

Canonical, the commercial sponsor for Ubuntu, plans to port the OS to the ARMv7 architecture, including the ARM Cortex-A8 and Cortex-A9 processor-based systems. ARM chips are used in many smartphones, such as Apple's iPhone.

ARM, based in the United Kingdom, has been making low-power processors for small devices much longer than Intel, but the market muscle of the latter makes the company a formidable opponent.

The building intensity of the competition was reflected last month in comments from Intel executives, who said the use of ARM processors in the iPhone was responsible for the device's Web-browsing shortcomings. Intel, which sells processors to Apple for use in its desktops and notebooks, apologized for the comments at its Developer Forum in Taipei and acknowledged that ARM processors draw less power than Atom.



In hopes of gaining more support for its products, ARM on Monday said it would collaborate with Adobe in making sure Adobe (NSDQ: ADBE)'s Flash Player 10 and other technologies could run on ARM chipsets in smartphones, netbooks, and other devices.

Source:

<http://www.informationweek.com>



COMMAND AND CONQUER

Written by Robert Clipsham

If you've ever misplaced a file, this month's Command and Conquer is just what you need. There's a bounty of graphical search tools out there, but why use a GUI when you can use the command line!

The first command we'll look at is grep. Grep is used for searching for a specified string within a file. For example, if I had a plain text file called cookies.txt with a recipe for making cookies in it, and wanted to find how many eggs I needed, I could do:

```
$ grep eggs cookies.txt
```

This will give a list of all lines that contain the word 'eggs'. As a more realistic example, say I had a configuration file (I'll use apache as an example here, though it could be anything) and wanted to change the name of the error log. Just knowing that it is in the file is no use, I need to know what line number to look at so I can change it.

```
# grep errorlog  
/etc/apache2/apache2.conf
```

Notice that I'm root when running this command. You may be able to run this command as your normal user, depending on the permissions for your configuration file. Try running it as a normal user first! You will notice that this command gives no output. This is because grep is case sensitive.

```
# grep -n ErrorLog  
/etc/apache2/apache2.conf
```

Using -n will make grep give line numbers, so you can find what line the directive is on. Notice that I have changed the capitalization of ErrorLog in this example. You could also use the -i option to make grep ignore case. You can also search all the files in a directory using -r:

```
$ grep -ir eggs recipes/
```

would find all recipes containing eggs in my recipes directory. What if you want to search by file name? This is where find comes in.

```
$ find recipes/ -type f -  
name '*.jpg'
```

This will find all the files (-type f) with a name ending in .jpg. Notice how I have enclosed *.jpg in single quotes? This prevents your shell from expanding the *. Try the following, and notice the difference.

```
$ echo *
```

```
$ echo '*'
```

The first should give a list of all the files in your current directory, while the second should give *. While find on its own may not seem that useful, when coupled with xargs it becomes a very powerful tool.

```
$ find recipes -type f -  
name '*-cake.txt' | xargs -  
I % cp % old-recipes/
```

This command takes the output of "find recipes -type f -name '*-cake.txt'", then pipes () it into xargs. Using -I % tells xargs to replace % with each line it receives.

```
$ find recipes -type f -  
name '*-cake.txt'
```



```
recipes/chocolate-cake.txt  
recipes/cheese-cake.txt  
recipes/fairy-cake.txt
```

If that (above) is the output, then these are the commands xargs will run:

```
$ cp recipes/chocolate-  
cake.txt old-recipes/
```

```
$ cp recipes/cheese-cake.txt  
old-recipes/
```

```
$ cp recipes/fairy-cake.txt  
old-recipes/
```

If you have a lot of files that match a certain pattern, this can be a very useful method to automate tasks such as a selective backup. If the output of find should be appended to the end, then you can remove -I % and it will be appended automatically to the end of the command. The final command that we'll cover in this issue is locate. Locate might not be installed, so you may need to install it before it works. Locate is a very fast way of finding files with a certain file name. If you have apache installed, try the following:

```
$ locate apache
```

A huge list of files will fly past, so it might be useful to combine its output with grep to find exactly what you're looking for.

```
$ locate apache | grep etc
```

This will narrow down the results to show only those containing 'etc'.



Robert Clipsham is a self-confessed geek, whose hobbies include: programming/scripting, chatting on IRC and not writing his articles on time.

Full Circle Magazine
The Independent Magazine for the Ubuntu Community

Ubuntu 8.10 is out! You can find information on the new release by [clicking here](#).

[New Thread](#)

Page 1 of 5 [1](#) [2](#) [3](#) > [Last »](#) ▾

Threads in Forum : Full Circle Magazine

[Forum Tools](#) ▾ [Search this Forum](#) ▾

The official Full Circle forum, hosted at Ubuntu Forums

<http://url.fullcirclemagazine.org/c7bd6f>



HOW-TO

Written by Elie De Brauwer

SEE ALSO:

[FCM#17 - Program In C - Part 1](#)
[FCM#18 - Program in C - Part 2](#)

APPLICABLE TO:

ubuntu kubuntu xubuntu

CATEGORIES:



DEVICES:



So far so good. This is already the third part in this series, and we have already made a lot of progress. The past articles explained how to think in code - by introducing functions. This article will probably be one of the most difficult in this series; here we'll touch something which is almost C-specific: the notion of pointers. C is a programming language which lives close to the

assembly language, and in assembly languages you spend most of your time moving data around, but in order to do that you need to know where the data is located, hence its address in memory. And this is just what a pointer is. A pointer is an address in your computer's memory, nothing more, nothing less. But when you're working with C, you'll end up with the conclusion that pointers are everywhere. So, after conquering functions and pointers, we should be able to handle almost anything.

In this article, I will not present a 'fully functional program'. I will present small snippets between the text, but you are encouraged to fire up your editor and start experimenting. You will also see that I introduce some 'extras' which are not mentioned in the main title. I will, for example, also introduce structures, arrays, strings, ... , because I want to see this series evolve into a practical tutorial, and not into a C textbook.

Two operators * and &

When handling pointers, you will encounter two 'extra' operators. These are * and &. It helps, when you look at code, that you read * as 'the value stored at this address', and & as 'the address of this variable'.

```
int anInt=5;  
int * anIntPtr=&anInt;  
  
printf("Address: %p Value:  
%d \n",&anInt, anInt);  
  
printf("Address of pointer:  
%p Address: %p Value: %d  
\n",&anIntPtr,  
anIntPtr,  
*anIntPtr);  
  
printf("Size of pointer: %d  
size of int: %d\n",  
sizeof(anIntPtr),  
sizeof(anInt));
```

Thus, we declare an integer and assign this integer the value 5, we declare a pointer (mind the extra *), and we let it point to the address of the previously



declared integer. Next, we print the address of the integer, and the value of the integer. Then we print the address of the pointer, the value of the pointer (which is an address, the address of anInt), and the value the pointer points to. And to end, we print the size of the pointers and the size of the integer. This produces the following output:

```
Address: 0xbfc819d8 Value: 5
Address of pointer:
0xbfc819d4 Address:
0xbfc819d8 Value: 5
Size of pointer: 4 size of
int: 4
```

Here we can see that both pointers and integers are 4 bytes large (which makes sense, since I'm on a 32-bit computer; if you run this on a 64-bit or a 16-bit computer, these values may vary). The address will be different on your system, but the fact that the address of the pointer and the address of the integer are only 4 bytes apart is not a coincidence; they are simply physically stored next to each

other. In printf, we use p to print a pointer (in hexadecimal), s to print a string (for more information see man 3 printf). The sizeof() operator used in the printf statement returns the size of an element (in bytes).

Handling arrays

What is an array? An array is simply a list of variables of the same type. In this example, we declare an array of integers where we can store 5 integers. At this point, we also declare how many integers we want to put in there (in this case five). Here we initialize the array at declaration, but we could do it elsewhere in the program as well.

```
int
anIntArray[5]={10,20,30,40,50};
printf("Address of array:
%p\n", &anIntArray);
printf("Size of array:
%d\n",sizeof(anIntArray));
for(i=0;i<sizeof(anIntArray)/si
zeof(int);i++)
{
    printf("Index:%x
Address:%p Value:%d Value:
%d\n", i, &anIntArray[i],
```

```
anIntArray[i],
*(anIntArray+i));
}
```

This code produces the following output:

```
Address of array: 0xbf8b55d4
Size of array: 20
Index:0 Address:0xbf8b55d4
Value:10 Value: 10
Index:1 Address:0xbf8b55d8
Value:20 Value: 20
Index:2 Address:0xbf8b55dc
Value:30 Value: 30
Index:3 Address:0xbf8b55e0
Value:40 Value: 40
Index:4 Address:0xbf8b55e4
Value:50 Value: 50
```

Now, what does this show us? The size of the array equals the number of elements times the size of each element (there is nothing extra stored). All elements are placed next to each other in memory (look at the memory addresses: they each differ by 4. By adding [i] after the array name, we can address an element of the array at index i. But, and here's some magic called 'pointer arithmetic', if we add 1 to an int pointer, the



pointer is increased by 4 (the size of the integer) - not by one. So, we can address the array by using the subscript method ([i]), but also with some pointer arithmetic, and, in essence, the array we declared is just a pointer to memory - where several values of the same type are stored.

Strings

We have touched on integers and arrays of integers, and we'll extend this principle. A single character ('c') can be stored in a 'char' type, and, if we take multiples of these chars, and put them after each other, a string is thus nothing more than an array of chars.

```
char aChar='c';
char * aString="Hello";
printf("Address: %p Value:
%c Size: %d\n",&aChar,
aChar, sizeof(aChar));
printf("Address of string:
%p\n", &aString);
printf("Size of string:
%d\n",strlen(aString));
```

```
printf("Value: %s\n", aString);
for(i=0;i<=strlen(aString);i++)
{
    printf("Index:%x
Address:%p Value:%c\n", i,
&aString[i], aString[i]);
}
```

Here, we create a char, and a char array (which is, in essence, a pointer; this is equal to writing 'char aString[6] = "Hello";', and do mind the difference between the char 'c' and the string "c"). This generates the following output:

```
Address: 0xbff8b560f Value: c
Size: 1
Address of string: 0xbff8b5600
Size of string: 5
Value: Hello
Index:0 Address:0x8048780
Value:H
Index:1 Address:0x8048781
Value:e
Index:2 Address:0x8048782
Value:l
Index:3 Address:0x8048783
Value:l
Index:4 Address:0x8048784
Value:o
Index:5 Address:0x8048785
Value:
```

There is actually nothing new here. We handle it the same way as adding integers, except we now use 'strlen()', a function defined in string.h (see man 3 strlen for details) to get the length of the string; a char is only one byte large, and we use %s to print it. There is only one magical thing here and that is how will we know that the string is finished? Well, the array is not {'H','e','l','l','o'}, it is {'H','e','l','l','o',0}. The ASCII null character is added after the string, so how does strlen() work? It is just a while loop which continues increasing the index until the value becomes 0.



This defines a structure...

Structures

Everything's going well. Let's add another thing on the pile - structures. We know arrays? Arrays are a collection of items of the same type; structures are a collection of things with different types.



```
struct aStruct
{
    int intMember;
    int * intPointer;
    char charMember;
    char ** stringPointer;
};
```

This defines a structure called 'aStruct', which combines an integer, a pointer to an integer, a char, and a 'double' pointer (a pointer to a string or a pointer to a pointer to a char). Put this declaration outside your functions. Typically, these are placed in header files. Next we can use this struct; we use the previously defined variables to populate this struct:

```
struct aStruct aStruct;
struct aStruct *
aStructPointer;

printf("Address: %p Size:
%d\n", &aStruct,
sizeof(struct aStruct));

printf("%p %p %p
%p\n", &aStruct.intMember,
&aStruct.intPointer,
```

```
&aStruct.charMember,
&aStruct.stringPointer);

aStruct.intMember=6;

aStruct.intPointer=&anInt;

aStruct.charMember='k';

aStruct.stringPointer=&aString;

aStructPointer=&aStruct;

printf("Member of struct:
%d\n",
(*aStructPointer).intMember);

printf("Member of struct:
%d\n",
(*aStructPointer).intPointer);

printf("Member of struct:
%d\n", aStructPointer-
>intMember);

printf("Member of struct:
%d\n", *aStructPointer-
>intPointer);

printf("Member of struct:
%s\n", *aStructPointer-
>stringPointer);
```

And the output:

```
Address: 0xbff8b55e8 Size: 16
0xbff8b55e8 0xbff8b55ec
0xbff8b55f0 0xbff8b55f4
```

```
Member of struct: 6
Member of struct: 5
Member of struct: 6
Member of struct: 5
Member of struct: Hello
```

And what does this teach us? Well, we can declare structures, we can have pointers to structures (it goes further, we can have arrays of structures, and structures can contain arrays, structures can also contain structures and structures can even contain pointers to structures of the same type -- this is called a linked list). By using the '.' operator we can access the members of a struct, and when we have a pointer to a struct, we do not need to dereference it first as in `(*aStructPointer).intMember`, since this is so common we can use the '`->`' operator as in `aStructPointer->intMember`. Also, using the double pointer is peanuts. There is, however, one odd thing in the output: here it says the size of this struct is 16, while we added one int (4 byte), one int pointer (4 byte), one char (1 byte) and one char pointer (4



byte). Who stole those three bytes of memory? Well that is called alignment. During the compilation process all memory addresses were aligned to 4-byte multiples since it is much more efficient for the processor to fetch an address which starts at an address which is a multiple of four. But if you would really want to change this, you can.

A word of caution

For all the brave who managed to bear with me this far, my congratulations. I know that the first time people talk about pointers it results in a lot of frowning and thinking 'why would somebody want to use this', but, don't panic - you just need a little practice to get full speed with pointers, and you'll soon see the advantage they bring. But one word of caution is in place: pointers point to 'a' memory location. They can point to any memory location. If you forget to initialize them, or forget to dereference them, you can end



You just need a little practice...

up in strange situations. I lost a day this week, because I incremented a pointer (which was zeroed afterwards) instead of incrementing the value the pointer pointed to. C will not prevent you from doing these things, but these will result in your application being terminated. It's the same with arrays: if you write `int array[5]; int b; array[6]=0;`, you will set the value of b to zero. This leads to memory corruption, and, in extremis, to stack corruption. So, pointers are very powerful, but you need to use them right.

Exercises

- Collect all the code snippets on this page and turn them into a working program.
- Try to run this program on a 32bit and a 64bit system (use a livecd for example), and compare the differences.
- Implement `strlen` yourself using a

while loop.

- Take a look at some manpages - those of `memcpy` `strcpy` `strcat` `memzero`, and see that all these functions operate on pointers.
- A C application typically has '`int main(int argc, char **argv)`' as its main prototype, here `argc` contains the number of strings passed to the application, and `argv` is an array of `argc` strings. Write a small application which prints all arguments given to the application. What is stored in `argv[0]` ?



Elie De Brauwer is a Belgian Linux fanatic, currently employed as an embedded software engineer with one of the world's leading satellite communications companies. Apart from spending time with his family, he enjoys playing with technology, and spends his days waiting for Blizzard to finally release Diablo III.



HOW-TO

Written by Rob Kerfia

SEE ALSO:

N/A

APPLICABLE TO:

ubuntu kubuntu xubuntu

CATEGORIES:



Dev Graphics Internet Multimedia System

DEVICES:



CD/DVD HardDrive USB Drive Laptop Wireless

Have you ever needed a temporary WiFi access point (AP), or just needed something until you can go out and buy one? This how-to should let you create one in a pinch.

First, make sure that your Wi-Fi card is well supported! For example, some Ndiswrapper-using cards might not work. Your mileage may vary, so check

CREATE A WIFI ACCESS POINT

<https://help.ubuntu.com/community/WifiDocs/WirelessCardsSupported>.

To start, we'll have to set your WiFi card to ad-hoc mode, which is a direct wireless connection between two or more computers with no router involved. Open your terminal on the AP (we'll be using it in this whole tutorial), and type:

`sudo iwconfig eth1 mode ad-hoc`

Replace eth1 with your wireless card's correct interface, if it's different. For example, some cards are ath0 or wifi0. If you don't know, you can run the iwconfig command, and it will display something like this:

```
ubuntu@server:~$ iwconfig
lo      no wireless extensions.

eth0    no wireless extensions.

eth1    unassociated ESSID:off/any
        Mode:Managed Channel=0 Access Point: Not-Associated
        Bit Rate:0 kb/s Tx-Power=20 dBm  Sensitivity=-8/0
        Retry limit:7 RTS thr:off Fragment thr:off
        Power Management:off
        Link Quality:0 Signal level:0 Noise level:0
        Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
        Tx excessive retries:0 Invalid misc:0 Missed beacon:0
```

In this case, the interface would be eth1.

Next, set the card to the channel you want the AP to be on. I set mine to 4, which will work for most people. Do this by typing:

`sudo iwconfig eth1 channel 4`

After this, we need to set an ESSID, which is basically the AP's name. Do this by typing:

`sudo iwconfig eth1 essid
APName`

We're almost done. If you want to have some encryption for your AP, you can add WEP encryption by typing:

`sudo iwconfig eth1 key
abcde12345`

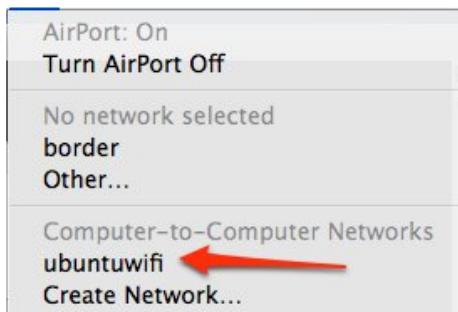
Replace abcde12345 with your preferred WEP encryption password.

Finally, give your access point an IP address with:



```
sudo ifconfig eth1
169.254.xxx.xxx
```

Now, go to another computer - the client, and check to see that the new AP is broadcasting. We will also set up Secure Shell (SSH) tunneling on this computer.



SSH Tunneling

Even though I'm using a Mac, it works exactly the same on Linux.

SSH tunneling is quite simple - all you do is create an SSH connection to another machine. You configure the local computer to forward all communications through that tunnel to the other machine. The other machine connects to the internet and sends the data. In this case, we're going to create an SSH tunnel through the ad-hoc connection (normally, there would only be

wireless access to the other computer, not the internet). Since the other computer is connected to the internet, we can tunnel communications through that machine, and connect to the internet wirelessly, without a router.

To create the tunnel, first check that SSH is started on the AP (the one that we did all the iwconfig stuff on). Just to be sure, type:

```
sudo /etc/init.d/sshd start
```

Now, go to the other machine - the client - (I'm using a Macbook Pro) and open a terminal window there. Remember we gave the other machine an IP address with sudo ifconfig eth1 169.254.xxx.xxx a while back? Now, we need it. Type:

```
ssh -ND 9999
youruser@169.254.xxx.xxx
```

to start the proxy. SSH will ask you something like:

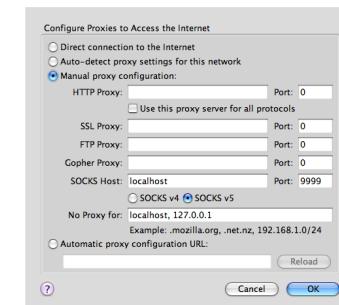
The authenticity of host '169.254.xxx.xxx' can't be established. Are you sure you want to keep connecting (yes/no).

Answer "yes", and enter your password. It will look as though

```
[~]$ ssh -ND 9000 ubuntu@169.254.92.13
The authenticity of host '169.254.92.13 (169.254.92.13)' can't be established.
RSA key fingerprint is d6:7f:7e:1f:7f:6d:59:8c:d0:7c:6e:27:90:97:15:9a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '169.254.92.13' (RSA) to the list of known hosts.
Permission denied, please try again.
ubuntu@169.254.92.13's password:
```

nothing has happened, but the SSH tunnel is now working.

Finally, start up Firefox.



Go to the "Advanced" tab, and click on "Settings". Click on "Manual proxy configuration", and enter in "localhost" for "SOCKS Host". Enter in "9999" for the SOCKS Host port. Leave everything else blank or at its default setting. Click OK.

You should now be able to browse the web, wire-free!

For more information, see <https://help.ubuntu.com/community/WifiDocs/Adhoc>.



HOW-TO

Written by Ronnie Tucker

SEE ALSO:

FCM #12 - #17 : USING GIMP 1 - 6

APPLICABLE TO:

ubuntu kubuntu xubuntu

CATEGORIES:



DEVICES:



In this, the last part of this GIMP series, I'm going to create a single image using three photographs taken from Flickr. Although the final image won't fool a photographic analyst, its creation should help you practice what you've learned in the previous seven articles. What I've also done is recorded, in real-time, my thirty-minute creation of the image, which you can view on

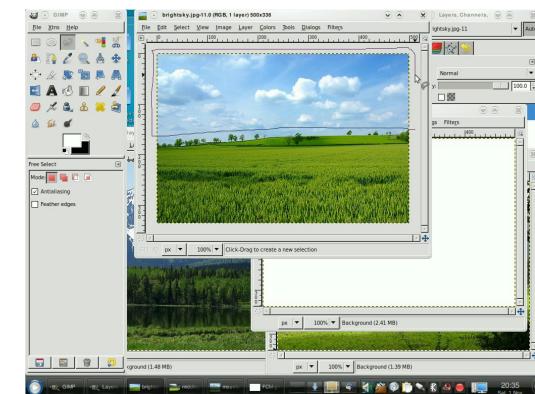
Google Videos at:

<http://url.fullcirclemagazine.org/c44b86>

First, I grab my three source images from Flickr, being careful to choose only photographs licensed under the **Creative Commons ShareAlike** license - which allows editing of photographs. What I'm thinking is that I'll use the sky from one photo, a middle-distance from a second photo, and possibly a foreground from a third photo, either that or create some water using GIMP filters. I'll post links to the source images at the end of the article, should you want to follow along.

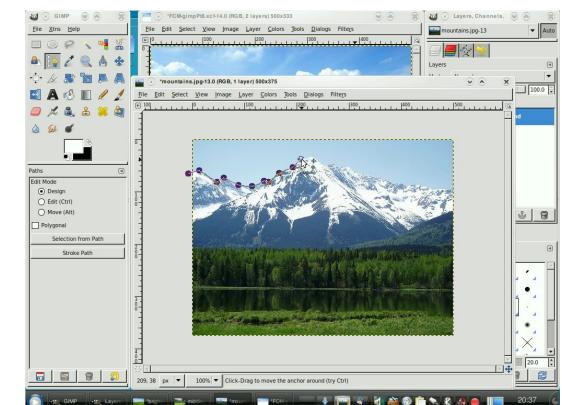
With my three images open, I create a new image of the same size as the source images. I used the Flickr preview images as I didn't want to have too many large images open while recording the screen, but you can use the full-size photographs. In the case of a landscape, it's better to work from back to front with the sky being farthest back, so I draw a very rough selection line around the sky of

USING GIMP - PART 8



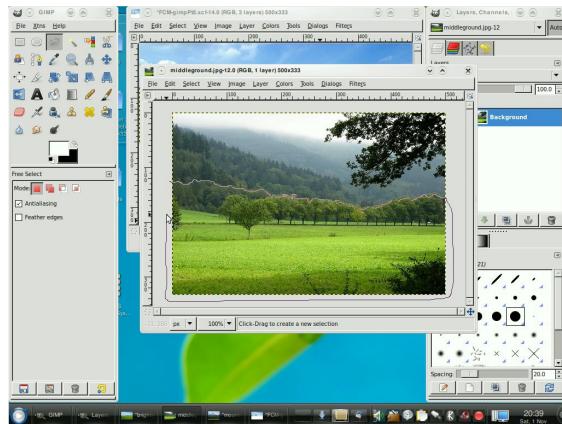
my first photograph. I copy that selection, and paste it into my new image.

I then decided to use the mountain range from the second photograph, and selected its outline using the Pen tool. Again, I copy and paste the selection into the new image.

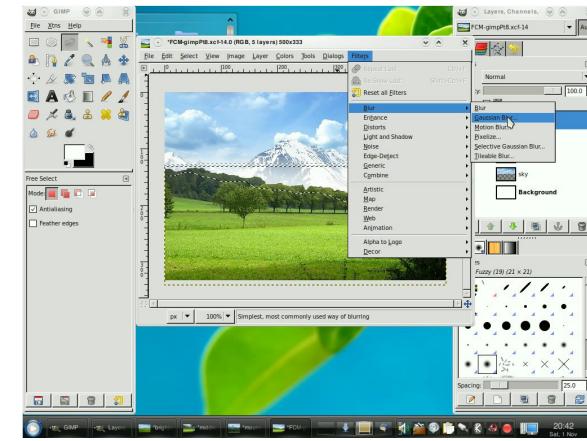
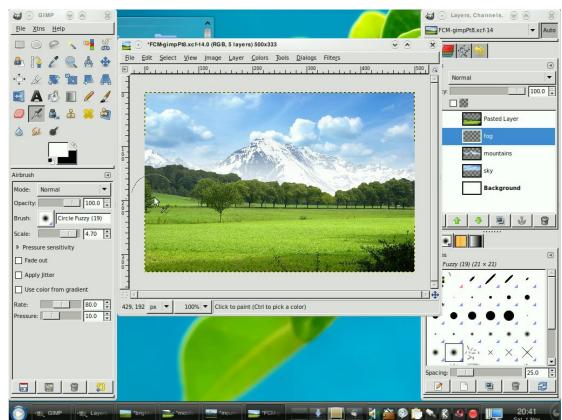




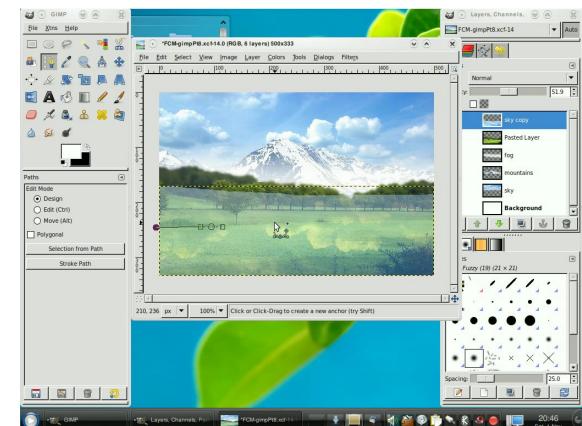
From the third photo, I select the trees and grass area and copy/paste it into the new image.



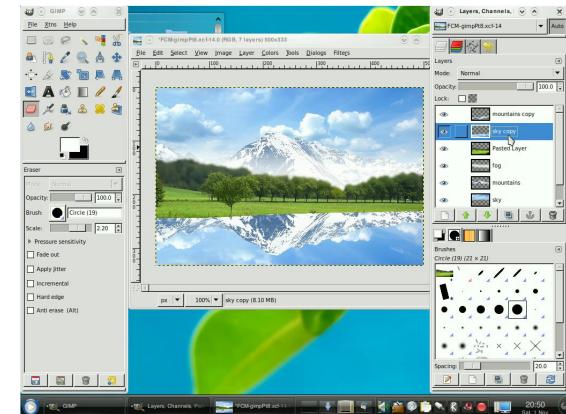
To add some depth to the image, I created a new layer and placed it between the mountain and tree layers. I airbrushed in some white to act as a fog/mist. I also selected the furthest away trees and applied Gaussian Blur to them.



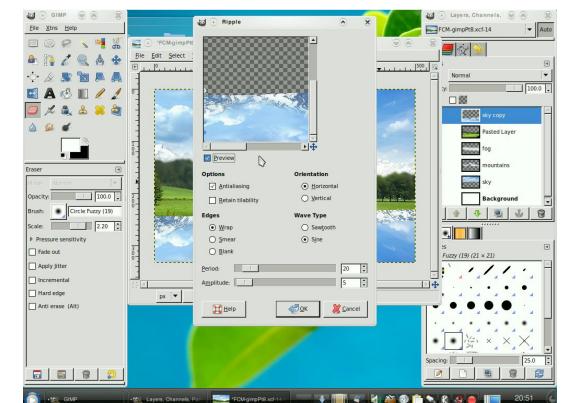
I had a fourth image that I was going to use, but decided against it. Instead I copied the sky layer, flipped it vertically, made a selection with the Pen tool, and erased parts of it. This will have effects applied to it and will become water.



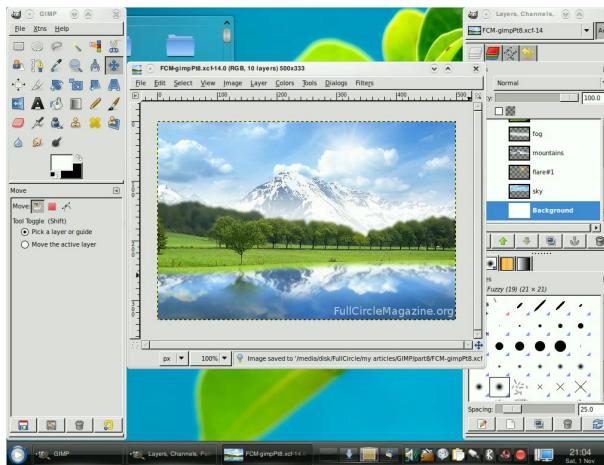
The mountains would also be reflected, so I did the same with the mountain layer: copy it and flip it vertically.



Before applying a ripple to the water, I used a soft-edged brush with the eraser to soften the outline of the mountain. Then I flattened the two layers (mountain reflection and sky reflection) into one layer and applied a ripple to them.



I darkened the foreground of the water using the Burn tool. At that point, I realised that I hadn't reflected the foreground tree! So



I did a quick selection of the foremost tree, copy/pasted it to a new layer, flipped it vertically, and applied a ripple effect to it. Thankfully, the ripple filter keeps your settings from the last time it was used, so the ripple on the tree was the same as on the water layer.

The components of the final image are now all in place. At this point, I began playing around with various filters to see if any would enhance the image. I tried Lens Flare and Sparkle but neither did anything effective. I tried the Gradient Flare, on a new layer, and it gave the effect of a sun, so I kept that and played with the layer effects to make it blend in more.



video of the above image being created: <http://url.fullcirclemagazine.org/c44b86>

Source Images:



<http://url.fullcirclemagazine.org/2fba72>



<http://url.fullcirclemagazine.org/8a00d5>



<http://url.fullcirclemagazine.org/d31f07>

I hope you've enjoyed this series of articles on using GIMP. Next month, we hope to have a new series of articles for you.



Ronnie Tucker is Editor of Full Circle magazine, a recent GNOME convert, and artist whose gallery of work can be seen at www.RonnieTucker.co.uk.



HOW-TO

Written by Italo Maia

SEE ALSO:

N/A

APPLICABLE TO:

ubuntu kubuntu xubuntu

CATEGORIES:



DEVICES:



A couple of months ago, I bought myself a nice new cellphone. It has a good camera, and a fine display. It's neither too big nor too small. It plays video and audio well. That's a good thing, since I'm a video-on-the-phone kinda guy. Especially music videos. I'm also an Ubuntu guy - to the bones. And I needed a program that could convert my music videos into a format my cellphone would accept.

My Nokia 6120 only plays mp4 and 3gp videos, so, my goal was set: find a Linux tool capable of converting my music videos from a wide range of formats to something my cellphone could play without glitches. After a quick search, I found a few tools that promised this, but they were all too hard to figure out. A friend of mine tried as well, but gave up very quickly. But not !! After a long and difficult search, I found what I was looking for: an easy to install, easy to use application, capable of converting my videos to formats compatible with my Nokia. Actually, what I had found was much more than just that.

I had found MP4TOOLS. It is capable of converting AAC audio and AVI/MPG video to formats compatible with the PSP, iPod, and Symbian. It has it all, and with a graphical interface to boot. The install process is easy.

In your Ubuntu System menu, go to Administration > Software Sources. Go to the third party

CREATE MOBILE MULTIMEDIA

software section, and add the two following repositories:

deb

<http://ppa.launchpad.net/teknoraver/ubuntu> hardy main

deb-src

<http://ppa.launchpad.net/teknoraver/ubuntu> hardy main

Now reload your apt-get list:

`sudo apt-get update`

and you're ready to install. Run the command:

`sudo apt-get install
mp4tools`

That is pretty much it. Now you have a set of tools for converting all your videos to nice quality, good looking, mobile videos - the easy way! The list of commands available to you are:

mk3gp encodes a standard 3gp file

mks60 encodes a high quality



3gp file (may not play on older phones)

mkamr encodes an AMR file (ringtone)

mkmp4 encodes a highest quality H.264 Video file, for PC playback

mkipod encodes a movie for the Apple iPod

mpps encodes a movie for the Sony PSP

dvd23gp rips a DVD to a standard 3gp file

dvd2s60 rips a DVD to a high quality 3gp file

dvd2psp rips a DVD to a movie for the Sony PSP

dvd2ipod rips a DVD to a movie for the Apple iPod

To convert any video (compatible with mp4tools) to your preferred format, simply use one of the

commands listed above, with the full path to your video file. For example, to convert `videofile.avi` to a standard 3gp file, use the command:

`mk3gp path/to/videofile.avi`

This will convert your `videofile.avi` to, in this case, a 3GP file ready for viewing on most mobile phones.



CONTACTING FULL CIRCLE MAGAZINE



Web: <http://www.FullCircleMagazine.org>

Forum: <http://url.fullcirclemagazine.org/c7bd6f>

email: letters@fullcirclemagazine.org

IRC: [chat.freenode.net #fullcirclemagazine](irc://chat.freenode.net/#fullcirclemagazine)



MY STORY

Written by Victor Moisey

My colleague, John, and I work for the Department of Geological Sciences at the University of Cape Town (UCT). Our responsibility is to maintain departmental analytical instruments.

One day, John told me that he had an old computer, and was interested in putting it to some use - but was undecided. I suggested Ubuntu.

The machine is a 500MHz Celeron. It had 128M of RAM, so I installed an additional 128M. Fortunately, I had a couple of 64M DIMMs lying around. The graphics card was troublesome, so out it came and I plugged in a spare. I couldn't say what card it was; all I know is that it worked.

I started the installation by running Hardy Heron Live. After satisfying myself that all was working, and after partitioning manually, I started the installation.

Being on the old side, the machine was slower than what we are accustomed to, so I left it to complete the installation.

At UCT we are fortunate to have the whole Ubuntu repository available on the intranet. The advantage is that 1Mbit download rates are possible. Internet access, on the other hand, is around 3 to 10k bits per second.

All I had to do was point the machine to the Ubuntu repository by changing sources.list appropriately. Then I started the update manager and followed that with installation of programs not included on the Ubuntu CD.

The result of all this work was a very happy John. I showed him what the machine could do, and we compared it with the time when that other operating system was installed. He

UBUNTU JOHN

was so keen to try it out that the photograph shows him trying it out before I had time to replace the cover.

He is interested in Scribus, Qcad, Open Office, and some games.

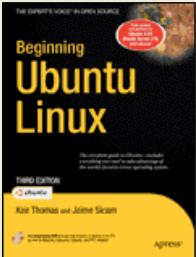
John is now a satisfied user of Ubuntu. He has managed the transition from that other operating system to Ubuntu with ease.





BOOK REVIEW

Written by Ronnie Tucker



By: Keir Thomas , Jaime Sicam
ISBN10: 1-59059-991-8
ISBN13: 978-1-59059-991-4
768 pp
Published: Jun 2008
eBook Price: \$27.99 (~£18)
Price: \$39.99 (~£25)

DVD: Ubuntu 8.04 LTS, also Kubuntu, Edubuntu, Xubuntu and PPC releases.

The book begins with a brief discussion about what Linux is, where it came from, why Windows isn't the best choice for an operating system, and the benefits of using Linux instead of Windows. Before moving on to the installation of Ubuntu, there is an explanation of the history and politics of Linux.

It helpfully explains how to backup any important files you may have in Windows before you install Ubuntu. Then it gives a step-by-step installation of Ubuntu, thankfully spending

several pages explaining partitioning, always a tricky step for new users.

The next chapter is on potential installation problems. Presenting this material early in the book could discourage some readers from trying Ubuntu, whereas a later presentation should not. The chapter Booting Ubuntu for the First Time is important for new users, and Beginning Ubuntu Linux explains it well with a very helpful table giving the Ubuntu equivalents for various Windows items such as 'My Computer.' An excellent idea.

Other too-early presentations in the book are the chapters on hardware configurations (using ndiswrapper, PCI IDs, console commands, and other technical processes), virus scanning, and security. Though important, they should really come after the user has had the chance to play around with Ubuntu.

Back to the good ideas: there is a chapter that lists applications, such as OpenOffice, GIMP and Firefox, and explains that for every Windows application there truly is a capable equivalent in Ubuntu. A table listing DOS commands and their Linux equivalents is another excellent idea. The many directories within Linux are also well explained, and are given brief descriptions of what's in these mystical places!

Email set up is explained using Evolution, and instant messaging applications are also touched upon. Audio and video codecs are well explained, and readers are given a quick lesson on why some audio and video files won't open without a few initial extra steps. I don't agree with the authors' advice to install RealPlayer, as it's not commonly used nowadays in Ubuntu. Speaking of audio, it



was good to see Magnatunes and Jamendo being mentioned.

Rounding off the multimedia presentations, it discusses GIMP, but not in depth. However, it gives enough information to help the new user (or Photoshop user) come to grips with the program.

The final chapters discuss things such as remote access and openSSH. While probably too complicated for Ubuntu novices, these discussions should be useful to readers beyond the novice stage.

One big downside to this book, and to similar books, is the absence of a detailed presentation of KDE -- Kubuntu is just briefly mentioned. I realize that a book presenting both GNOME and KDE would be around three inches thick, and be too heavy to lift, but still... not even a mention of Amarok, Kmail, or K3B?

All in all, this is an excellent book. Its title says 'Beginning Ubuntu Linux', and, although it is for beginners, it still has enough

information in it to satisfy the intermediate, and possibly even the more advanced, user.

To win a copy of **Beginning Ubuntu Linux - Third Edition**, answer this simple question:

Which email client is used in the Beginning Ubuntu Linux book?

Email your answer to:
competition@fullcirclemagazine.org

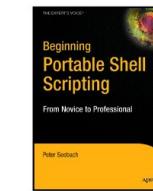
A winner will be selected at random on Saturday 27th December at 1600 UTC, and announced in **FCM#20**, which will be released on Sunday 28th December.

Good luck!

A big thank you to Apress for supporting Full Circle with the review copy of **Beginning Ubuntu Linux**, and for allowing us to use it as a competition prize.

FROM THE DESKTOP TO THE NETWORK

LOOK TO APRESS FOR ALL OF YOUR OPEN SOURCE NEEDS

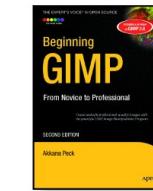


Peter Seebach
978-1-4302-1043-6
\$34.99 | 300 pp | November 2008

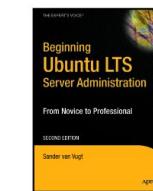
Andy Channelle
978-1-4302-1590-5
\$39.99 | 450 pp | December 2008



Akkana Peck
978-1-4302-1070-2
\$49.99 | 584 pp | December 2008



Keir Thomas & Jamie Sicam
978-1-59059-991-4
\$39.99 | 768 pp | June 2008



Sander van Vugt
978-1-4302-1082-5
\$39.99 | 424 pp | September 2008

Sander van Vugt
978-1-4302-1622-3
\$44.99 | 400 pp | December 2008



Apress books are available at many fine bookstores worldwide.

Don't want to wait for the printed book?
Order the eBook now at [http://eBookshop.apress.com!](http://eBookshop.apress.com)

Apress®
THE EXPERT'S VOICE™



MOTU INTERVIEW

Taken from behindmotu.wordpress.com



EMILIO P. MONFORT

Behind MOTU is a site featuring interviews with those known as 'Masters of the Universe' (MOTU). They are the volunteer army of package maintainers who look after the Universe and Multiverse software repositories.



Age: 19

Location: Murcia, Spain

IRC Nick: pochu

How long have you used Linux, and what was your first distro?

I started with Ubuntu in December 2005 with Ubuntu Breezy Badger, and have been using Linux since then. I mostly use Ubuntu, but I have a Debian VM for testing the packages I maintain there.

When did you get involved with the MOTU team and how?

I started contributing back in January 2007 with small packaging fixes, adding/fixing desktop files. I had no programming skills by that time (and they are still very poor, although I'm working on that!), so I started with simple tasks, and I've been taking more complex things as time passed. I've concentrated in some packages too, and I maintain some in Debian and Ubuntu.

What helped you learn packaging and how Ubuntu teams work?

The #ubuntu-motu channel in IRC has always been a great help, with a lot of MOTUs and other wannabes hanging there, and willing to answer questions or point to the proper documentation.

Also, the wiki, the PackagingGuide, and the Debian Policy, have been good places to consult when I didn't know anything, or was unsure.

But the most important thing has been to do packaging. You can read a lot of documentation, but won't learn packaging until you start working on it.

Favorite part of working with the MOTU?

The fact that many MOTUs are volunteers motivates me. Also the environment, and the good work the team does in a universe of thousands of packages.

Any advice for people wanting to help out MOTU?

Start contributing. You don't



need to know programming. You don't need to know packaging. You don't even need to know what that means! Just have a look at <https://wiki.ubuntu.com/MOTU/GettingStarted> and <https://wiki.ubuntu.com/MOTU/TODO>, and start contributing. And don't hesitate to ask in #ubuntu-motu, on IRC or in the ubuntu-motu-mentors mailing list, if you have any questions, even if you think they are obvious.

You have been working on a lot of different packages in the last cycles, what are you going to focus on in Intrepid?

I'm contributing to the DesktopTeam, merging gstreamer related packages, and looking at GNOME packages I'm interested in. I'm also looking at the SponsorsQueue from time to time, and do some reviewing.

Also, I don't want to duplicate efforts, so I've been working with Sebastian Dröge in having the GStreamer stack in sync with Debian.

For the Intrepid cycle, I want to

keep the GStreamer packages in sync with Debian, and try to reduce the duplication work between the Debian and Ubuntu GNOME teams.

How do you think Intrepid will be special for our users?

With GNOME 2.24, it will be a very solid release. It will be a pity it's not LTS.

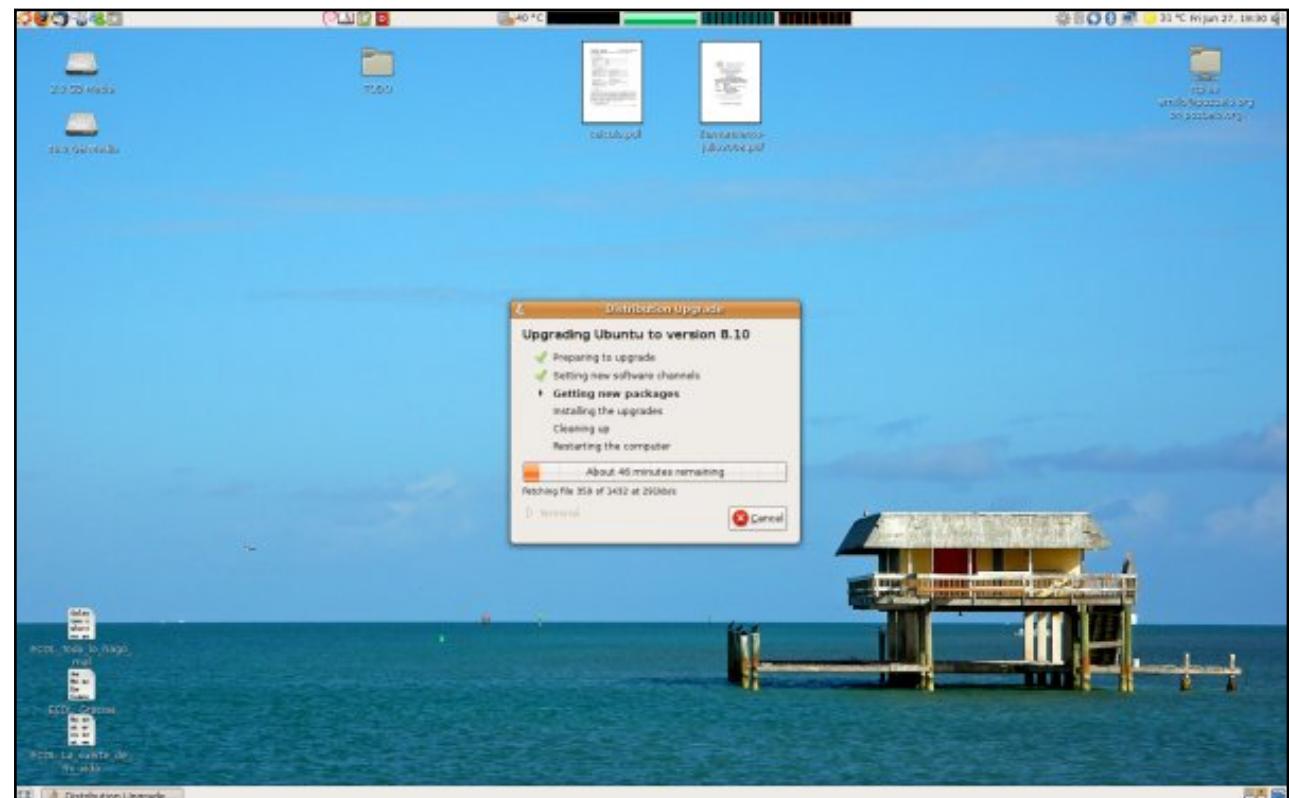
Favorite quote?

I don't have a favourite one, but I like this one a lot:

"A real friend takes your hand and touches your heart"

What do you do in your other spare time?

I like to go to the cinema, to meet my friends, and to watch the tv.





LETTERS

I'd like to know the status of the Ubuntu Home Server project. It seems to have died. If not, I would love to see a story updating its status and getting some buzz going about it. I'm surprised that no one thought to do this before Microsoft. It seems to be a perfect way to have Linux get a foot in the door of home users. Also, the Mint coverage is great. It seems like a good "flavor" to try.

Jeff Taylor

Ed: I checked their wiki pages and they haven't been updated in a while now, but I'm hoping that means the developers are beavering away on the project. We'll try and cover this again soon.

Thank you for your support. I am now writing this on a Linux machine, most of the problems I had are now gone.

Every month we like to publish some of the emails we receive. If you would like to submit a letter for publication, compliment or complaint, please email it to: letters@fullcirclemagazine.org. PLEASE NOTE: some letters may be edited for space reasons.

LETTER OF THE MONTH

Writer of Letter of the Month wins two metal Ubuntu case badges!



A few weeks ago I had to send a client some sketches which I only had paper copies of. Looking around I found **gscan2pdf**. I scanned the pages in and produced a multi-page PDF containing both landscape, and portrait, pages depending on the orientation of the drawing. I did not have to save any JPEG files or use OpenOffice, just one program from scanner to PDF. I believe it will also import images if required [it does indeed - Ed]. You may

want to do a follow-up article on using this program as an alternative method.

I could not get 'help' to work, but the icons seemed to do what I expected.

Anthony Corriveau

Ed: Many thanks for that Anthony, that helped me import, and convert, some JPEG images to PDF's for easy printing.

banking is now done with GNCash.

So, you see, even a Windows user like me can manage the switch.

Gerhard Schulze



UBUNTU WOMEN

Written by Elizabeth Bevilacqua



As Ubuntu Local/Community (LoCo) teams grow, many teams find that an overwhelming number of their members are male. Some teams even report that they have no active females at all!

How do you reverse this trend in your team? The first step is wanting to. The best argument for wanting to attract more women is that, since women make up over half the world's population, not attracting women to your team means you may be missing a vast untapped resource! Also, a more diverse team may open doors to projects that your team had never dreamed of.

Traditional Linux User Groups have faced this dilemma for years, and in that time the following suggestions have proven successful:

- Hold your meetings and events in safe, public locations.
- Adhere strictly to the Ubuntu Code of Conduct.
- Treat new members with courtesy, encourage their participation, and invite them to speak up about their current skills and interests. Extend this same treatment to women who attend.
- Don't underestimate or patronize girlfriends and wives - a woman might be attending a meeting or event with a partner, but she may surprise you with her enthusiasm for the project and/or technical expertise.
- Steer clear of sexist and other exclusionary jokes and stereotype-ridden comments, and avoid addressing your team with male-specific terms such as "fellas"[sic] and "gentlemen".
- Encourage women already

involved in your team to take visible roles within the team. Their visibility may encourage other women to step up and get involved.

Don't get discouraged if change doesn't happen overnight. There are still fewer women than men involved with Linux worldwide, and it will take some time to change this. In the meantime, if you make women (who are already involved) feel welcome and included in your team, you will benefit from their skills and talents. In addition, not only will the practices described above attract women, but they will attract other people also. Your LoCo will become a shining example of the inclusiveness that I believe Ubuntu wants to exude.



Elizabeth Bevilacqua is a Debian Systems Administrator in Pennsylvania and has been using Linux since 2002. She currently leads the Philadelphia Linux Users Group (PLUG) and the Philadelphia Chapter of LinuxChix (PhillyChix).



UBUNTU GAMES

Written by Edward Hewitt

GAME NEWS

- New EVE Online expansion announced - Quantum Rise is a new expansion of the MMORPG, EVE Online. The new expansion boasts new content and performance enhancements.
- Prey (below) coming to Linux - Yet another software developer is porting its games to Linux! Prey is a FPS game from 3D Realms. The demo is available now!



Open Arena is the open source equivalent of Quake 3 Arena. Like Quake, Open Arena is a first person shooter that is heavily based around multi-player deathmatch arenas. You fight it out in arenas, either online or against AI-controlled bots. There are several weapons you can use, such as the usual machine gun, shotgun, rocket launch and rail gun. There are 4 modes you can play: deathmatch, team deathmatch, capture the flag, and tournament.

As soon as you install the game, you have the option of playing the game in either single player mode or multi-player mode. The single player mode is good for players who are new to the game. It gives you a chance to get used to the fast-paced gameplay, and explore the many arenas. The AI-control bots are fairly intelligent and can put up a good fight. There are about 20 maps which you can play through. These are the



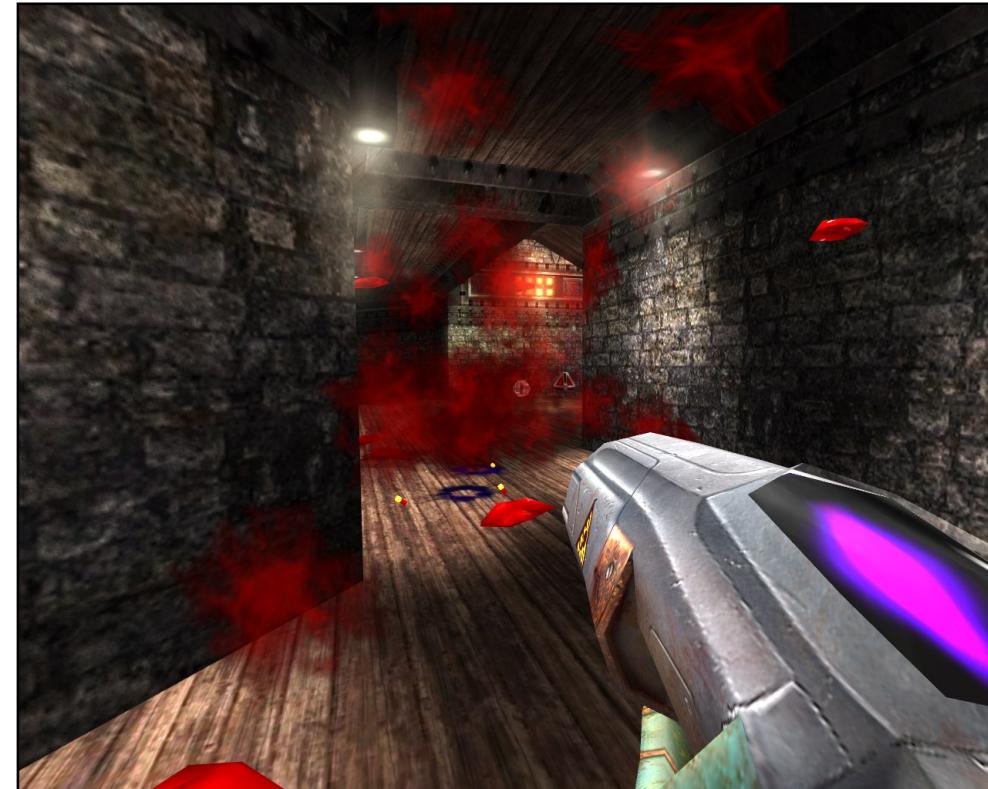
same maps that are used on multiplayer, so the single player mode gives you the chance to explore each map before you play online.

Finally, the multi-player - the main part of Open Arena. This game, like Quake 3, was designed with a focus on the online mode. You can play either over a LAN or over the internet. When you join a match, it is incredibly fast-paced. Stand still for a second, and your head will be shot off. Grab a gun, run and



fire. Open Arena requires no real tactics; unlike Counter Strike, you just have to go with the flow. There are about 70 servers for Open Arena, which are all fairly active. Each server has different maps to play on, and different game modes. For players new to online FPS games, it is not the best game to start out with. Many players are very fast, and have very good aim. I was on one server, where every time I spawned I would be shot by a rail gun. This happened 20 times in a row!

Open Arena has an active community of developers, always busy developing new versions of the game. In the past, they have added new game modes, new graphics, and new maps. The latest version of the game is 0.8.0. However, on the Ubuntu repos, the latest stable version is 0.7.7. I would recommend installing 0.8.0, because it has all the new content, and most of the



Open Arena servers will be using that version. You will be able to download the DEB file from PlayDeb.net.

Open Arena is an excellent free alternative to Quake. It is very well developed, and the game is perfectly balanced. Graphics are very dated, but with most online FPS games, it's more important to have a high frame rate than good looking graphics. I would strongly recommend this game to any hardcore Linux gamers,

especially FPS fans. However, I would not recommend this to gamers new to FPS, it is just far too fast.

System Requirements

Intel Pentium II 233MHz
64MB RAM
OpenGL Graphics Card with 32MB VRAM (3D acceleration enabled)
300MB HDD space



Ed Hewitt, aka chewit (when playing games), is a keen PC gamer and sometimes enjoys console gaming. He is also on the development team for the Gfire Project (Xfire Plugin for Gfire).



Q&A

Written by Tommy Alsemgeest

QI am trying to automount my second internal hard drive, but something is preventing it from mounting. I checked the properties of the drive that was manually mounted on my desktop, and in the details section for the other drive, I emulated what was in the information for the first one (/media/drive). Now, if I try to mount the drive, I get an error: can't mount due to unexpected character.

AEdit your fstab with the command

`gksudo gedit /etc/fstab`

and find your drive. If it isn't in the list, take a look here for instructions:

<http://www.tuxfiles.org/linuxhelp/fs-tab.html>

QMy update manager has two backports in it, and it won't let me check the box to install them. How can I

If you have Ubuntu-related questions, email them to: questions@fullcirclemagazine.org, and Tommy will answer them in a future issue. Please include as much information as you can about your problem.

install these, and why won't the "update manager" let me?

AThe backported programs probably have unresolved dependencies, so a program you have installed relies on one version of another program, but an update relies on a newer version. Thus, it can't upgrade the program without breaking another program. It's probably best to leave it alone until it is allowed to upgrade.

QI run 8.04 with a wireless card to a router in the basement of my house. A few days ago, I stopped being able to communicate with the router, and while I can get it going again, the network manager interface has changed. The network manager icon has become an icon showing two monitors. I can no longer access VPN sites, and I am unable to save WPA passwords. Also, when I click on the network manager icon, both options want the WPA password, but neither will save it.

ATry going to the network manager, selecting your wi-fi, and clicking properties. Make sure it is set on "roaming".

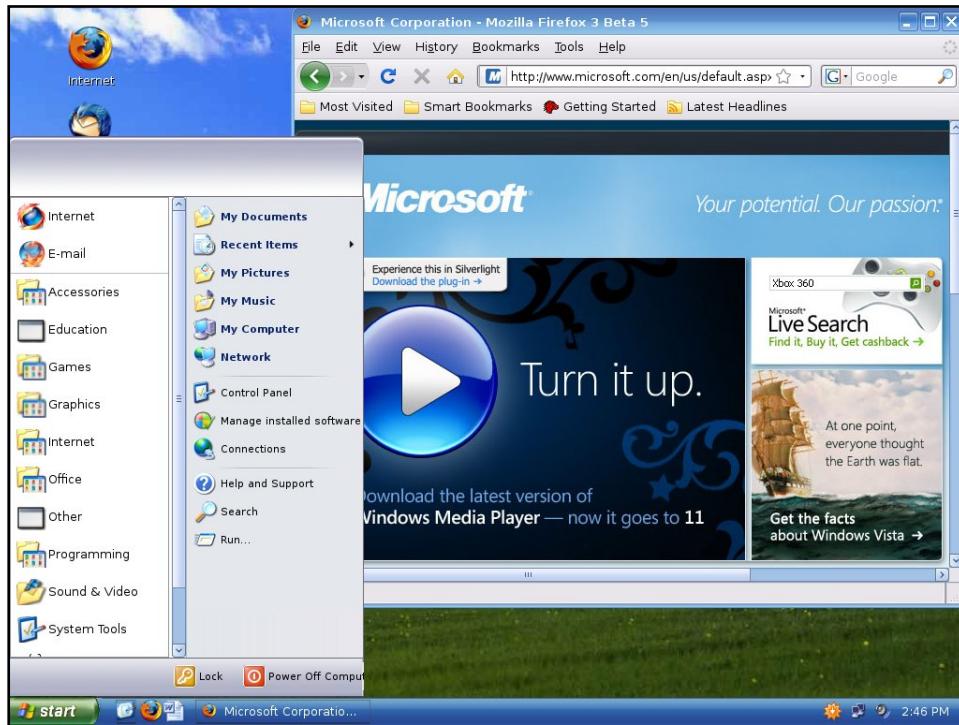
QI want my Ubuntu to run faster. I want to know if there are any folders which would make the system boot or run faster if they were held on their own partitions.

AI've heard people suggest /usr, /var, /home, and /boot. Also, place the partition on a separate hard drive, then it might speed it up. I doubt that having a separate partition on the same hard drive would help. It could even degrade performance, as the read/write heads have to move to separate sections of the disk to access the required files. Separate partitions would be good for protecting against file corruption, though not quite as good as separate drives.



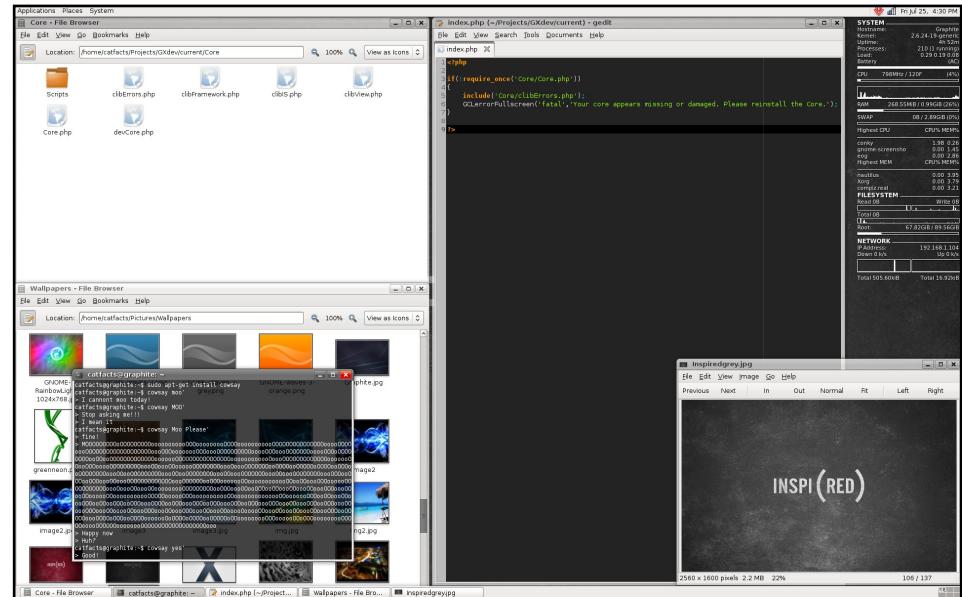
MY DESKTOP

Your chance to show the world your desktop or PC. Email your screenshots and photos to: misc@fullcirclemagazine.org and include a brief paragraph about your desktop, your PC's specs and any other interesting tidbits about your setup.



I've gone and made my desktop as much like Windows XP as possible. The wallpaper is Windows XP's Bliss, and the theme is Royale. I've installed a Windows-like start menu from Gnome-look.org, and I use Wine to install and run many Windows programs. The PC I'm using is an ASUS machine, with a 2.89GHz Pentium 4 processor and 512MB RAM.

Kevin Clotfelter

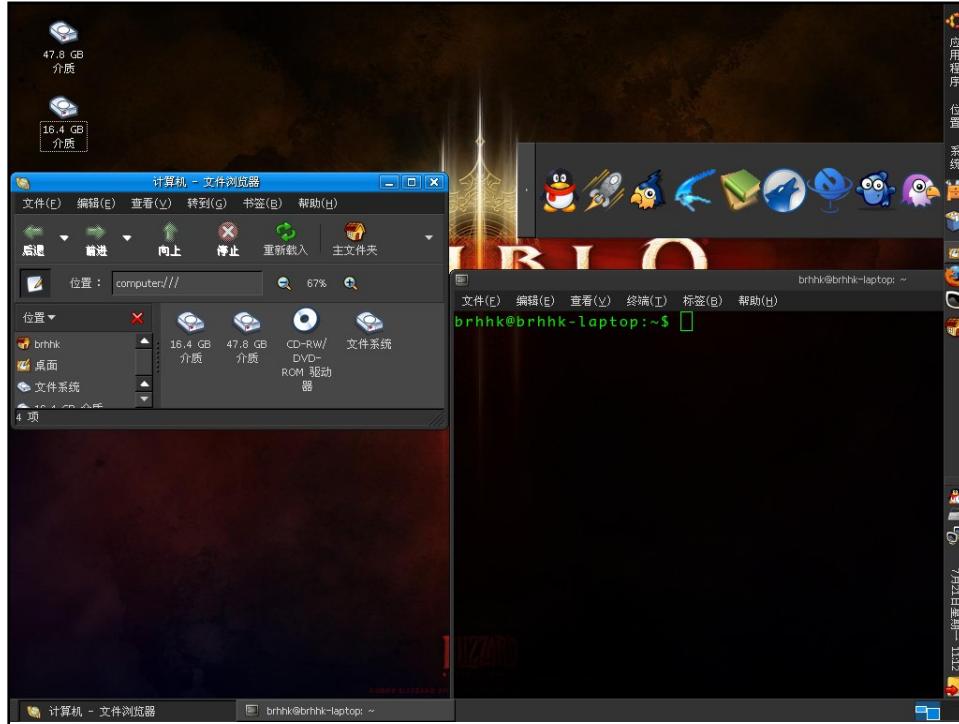


I'm running Ubuntu Hardy 8.4 with GNOME, using Compiz Fusion, and the Metacity, not emerald, window decorator. I have tried many dark themes, but they interfere too much with Firefox and applications, so I settled for this blend of the two. The background was originally red and I bleached it to fit with the greys. The Metacity theme is clearlooks, with a cherry on top (if I remember correctly), and the actual GTK is clearlooks too! The folders are Dropline NOU! and I use the dmz-black pointers. I've always liked simple but beautiful desktops, and this is more windows than I would ever have open on one workspace normally. Conky is usually not running (to save CPU cycles) - I turned it on specially for this screenshot.

James Savage

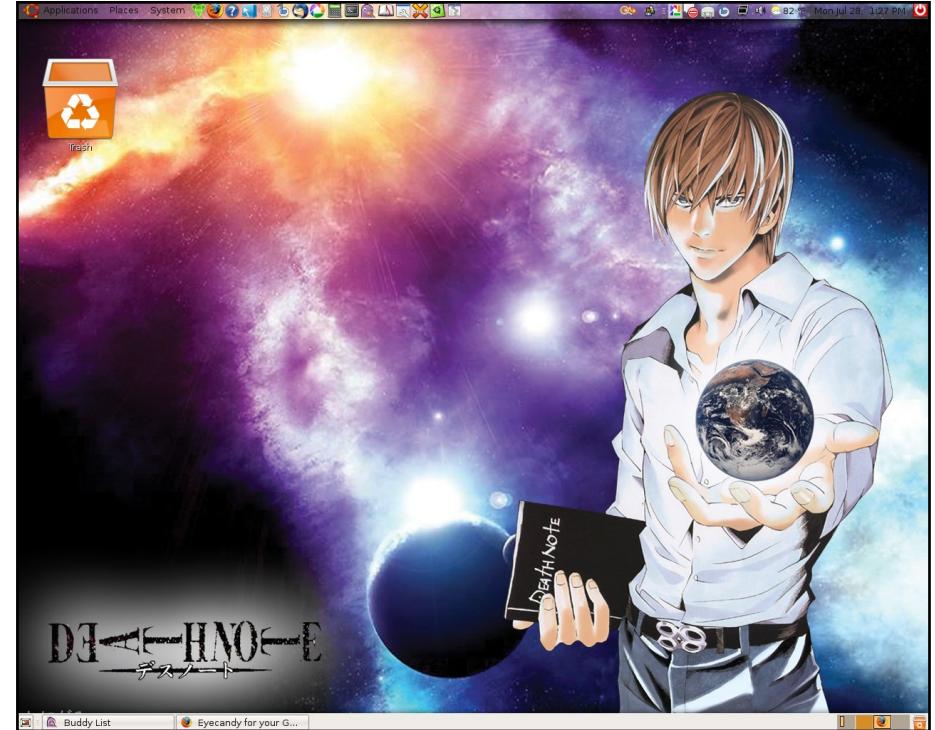


MY DESKTOP



I've used Ubuntu since 6.04 from the free live CD. There are few people using Linux in China. But I love it. I installed Hardy on my IBM R60 with Core 2 Duo, 1GB Ram and ATI x1300 video card. It runs perfectly with Ubuntu. Black, simple and clean is what I favor. I put almost all of my icons on the right, and all of my open windows on the bottom, and use all the desktop for my work. The penguin icon on the right is QQ, the most popular instant chat tool, like MSN, in China. It runs only on Windows, but I Wine it perfectly.

Brhhk



Well, since I found out about Ubuntu, I love it! I like it so much that, even though I still have Windows XP, I no longer use it. I think that Ubuntu Linux is what I've been looking for ever since I first used a PC and a Mac. Good job on Full Circle Magazine! I like reading it. Thanks for the info you guys put in it. I want to see how far Linux can go. I live by this Motto that I found on a wallpaper: "many choices, many flavors, FREEdom".

Nelvin



VIRTUALIZATION APPLICATIONS

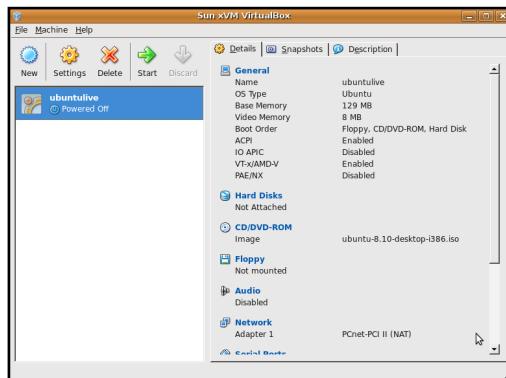
Written by Andrew Min

VirtualBox

<http://www.virtualbox.org/>

This VM (virtual machine) application, owned by Sun Microsystems and created by a small company called innotek, is one of the most popular virtualization solutions for Ubuntu. It's the third most popular method to run Windows apps on Linux, according to DesktopLinux.com (trailing Wine, which is not an emulator/virtualizer, and VMWare, which is proprietary). There's good reason: it has a bevy of features, including snapshots, shared folders, RDP, ability to use host USB, and a lot of advanced hardware virtualization.

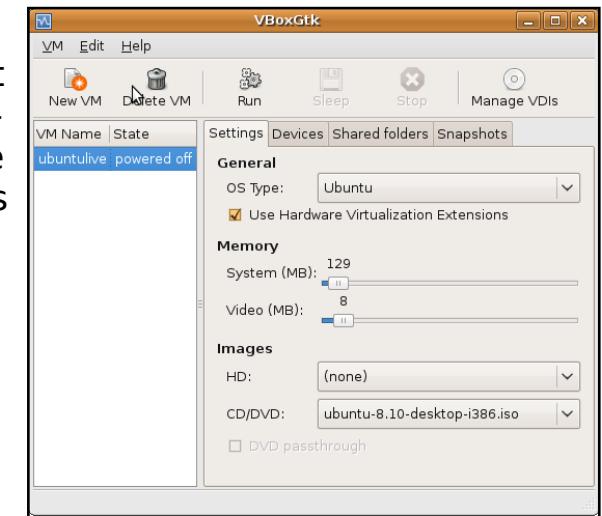
There are two versions of VirtualBox. Installing the open-source edition is easy: just install the **virtualbox-ose** package in the universe repositories. If you want to install the enhanced but closed-source version, you'll need to visit the website (above) and download the .deb.



VBoxGTK

<http://www.xente.mundo-r.com/narf/vboxgtk/>

If you're a Gnome or Xfce user who doesn't like using Qt applications on GTK-based systems, give VBoxGTK a whirl. It's still in beta (the site notes that "most of VirtualBox features are not supported, and those that are will probably fail"), but it's a good beginning and progressing rapidly. Right now, the developer is still focusing on stability, and as a result, most of the VirtualBox features aren't available. Still, it boots operating systems, and it's GTK.



To install VBoxGTK, use the **vboxgtk** package in the universe repositories.



Qemu Launcher



http://projects.wanderings.us/qemu_launcher

If you like the idea of QtEmu but dislike the fact that it's, well, Qt, Qemu Launcher is a good

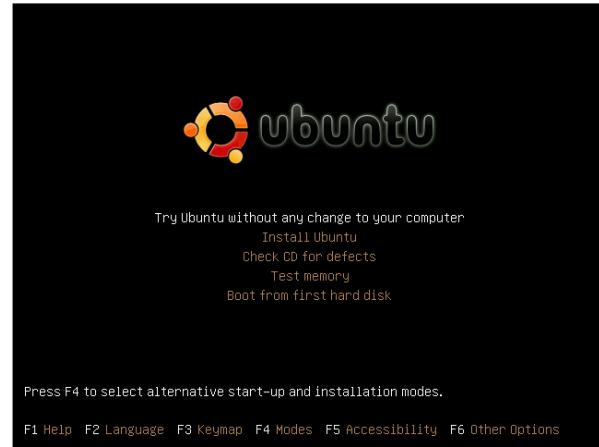
alternative. This GTK+ interface to QEMU supports almost all of QEMU's features, though it's not quite as user friendly as QtEmu. It's definitely hacker-targeted, with support for networking, acceleration, directly booting the Linux kernel, and some extra features like synchronizing the clock. Snapshots are also supported, making Qemu Launcher a viable alternative to the command line for hackers and programmers.

To install Qemu Launcher, use the **qemu-launcher** package in the universe repositories.



Andrew Min has been a Linux addict since he first installed openSuSE in VMWare. Learn more about him at <http://www.andrewmin.com/>

QEMU

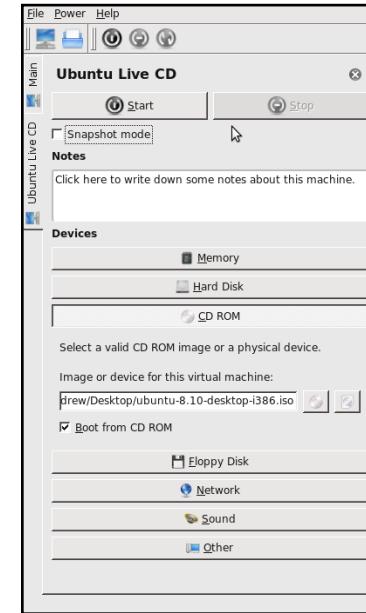


<http://bellard.org/qemu/>

While not the most user friendly application on the virtualization market, QEMU is one of the oldest and most respected virtualization programs around. QEMU can not only virtualize an existing chip (like VirtualBox or VMWare), but can also emulate a hardware platform like PowerPC or SPARC. This makes it really popular with developers trying to port programs to other platforms - like the Android. Plus, there are tons of handy features like snapshots, VNC, and of course standard features like sound, hard drive tweaking, and multiple CPUs. Code from it has been used in almost all virtual machines today, including VirtualBox.

To install QEMU, use the **qemu** package in the universe repositories.

QtEmu



<http://qemu.org/>

If you don't like messing around in the console, try out this Qt-based QEMU frontend. It's a straight-forward and easy-to-use GUI for QEMU that supports basically all the features of QEMU from a friendly Qt interface. There's all the standard memory, hard disk, CD ROM, Floppy, network, and sound options, but there's also some advanced features like seamless mouse integration and multiple CPUs. It's a sweet little app with a lot to offer, especially if you don't like Sun's dual-licensing of VirtualBox but love the interface.

To install QtEmu, you'll need the **qtemu** package from the universe.



HOW TO CONTRIBUTE

We are always looking for new articles to include in Full Circle. For article guidelines, ideas, and for issue translation, please see our wiki:

<http://wiki.ubuntu.com/UbuntuMagazine>

Please email your articles to: articles@fullcirclemagazine.org

If you would like to submit news, email it to: news@fullcirclemagazine.org

Send your comments or Linux experiences to: letters@fullcirclemagazine.org

Hardware/software reviews should be sent to: reviews@fullcirclemagazine.org

Questions for Q&A should go to: questions@fullcirclemagazine.org

MyDesktop screens should be emailed to: misc@fullcirclemagazine.org

If you have questions, you can visit our forum: www.fullcirclemagazine.org

FULL CIRCLE NEEDS YOU!

A magazine isn't a magazine without articles and Full Circle is no exception. We need your Opinions, Desktops and Stories. We also need Reviews (games, apps & hardware), How-To articles (on any K/X/Ubuntu subject) and any questions, or suggestions, you may have.

Send them to: articles@fullcirclemagazine.org

Full Circle Team



Editor - Ronnie Tucker
ronnie@fullcirclemagazine.org

Webmaster - Rob Kerfia
admin@fullcirclemagazine.org

Comms Mgr - Robert Clipsham
mrmonday@fullcirclemagazine.org

Proofreaders

Robert Orsino
Mike Kennedy
David Haas
Gord Campbell
Jim Barklow
David Sutton
Andrew Min

And our thanks go out to Canonical, the Ubuntu Marketing Team and the many translation teams around the world.

Deadline for Issue #20:
Sunday 07th December.

Release date for issue #20:
Sunday 28th December.