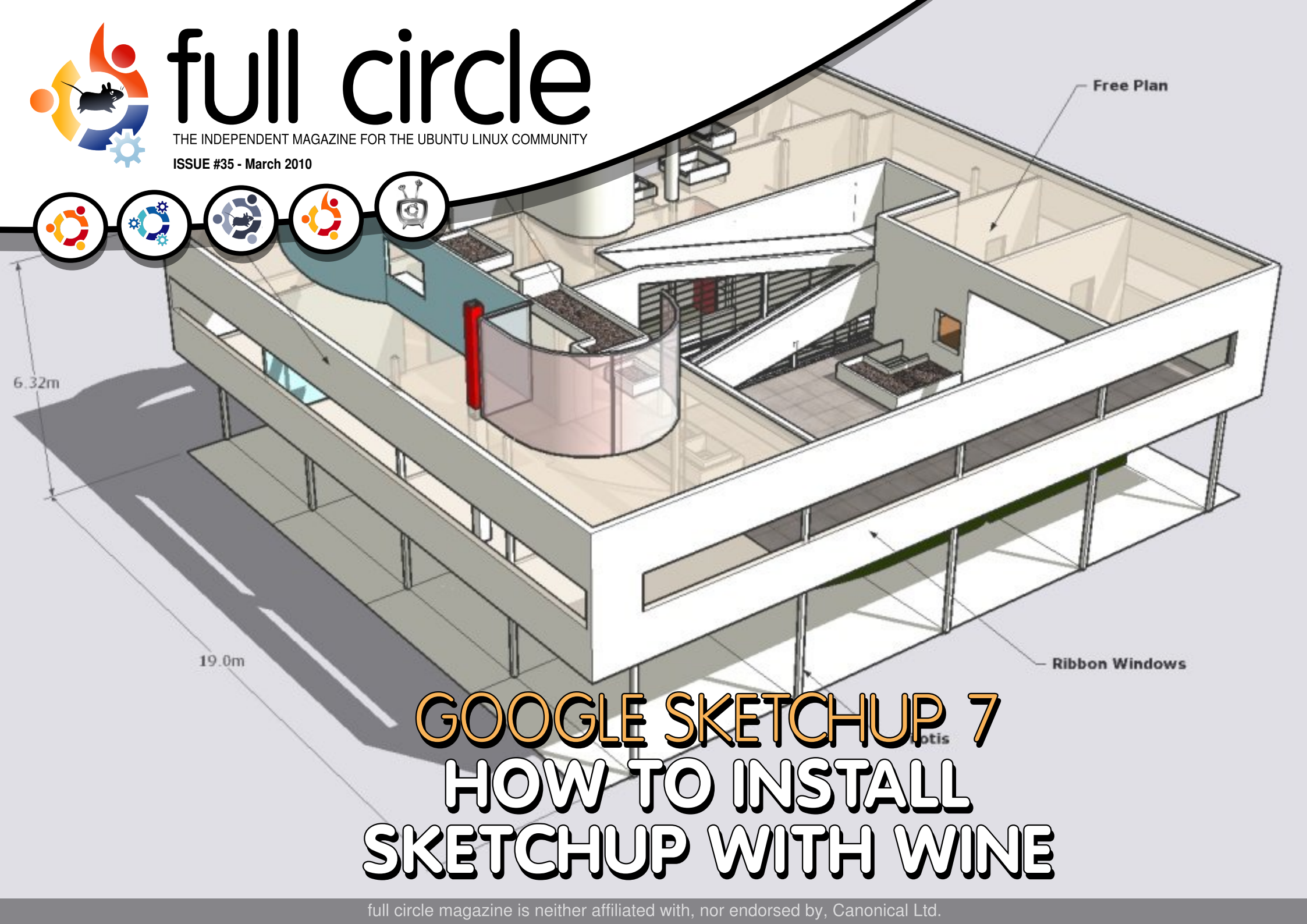
full circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

ISSUE #35 - March 2010



GOOGLE SKETCHUP 7

HOW TO INSTALL SKETCHUP WITH WINE

full circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

EDITORIAL

Welcome to another issue of Full Circle magazine.

Thie big news this month is, without doubt, the rebranding of Ubuntu. Both Mark Shuttleworth and Jono Bacon have announced that from Ubuntu 10.04 the brown is gone.

The first change is to the Ubuntu logo. Gone is the rounded font and tri-colour 'circle of friends'. In comes a sharper logo with an orange 'circle of friends' (below) which, in my opinion, gives the entire logo a fresh new look. Of course, it does mean I'll have to change the *Full Circle* logo to keep up with the times, but it won't be much, you'll still recognise it. I'll use the new Ubuntu font when it's released and change the colour of our version of the 'circle of friends'. Can we call ours the 'circle of readers'?

Another change to the Ubuntu look is the default theme. It is now called 'Light' and is available in both light and dark versions. I do think they're going a bit too close to the Apple theme's which I don't agree with. Surely we can come up with something better than Apple. But, at least the brown is gone.

As if all that wasn't enough, Ubuntu 10.04 Beta is now available! Check the following page for information and links. I, unfortunately, have to wait as I don't want to risk upgrading to 10.04 (from 9.10) until we release this issue. Then I can backup everything and upgrade without a care in the world.

Enjoy the issue and keep in touch!

All the best!
Ronnie
Editor, Full Circle magazine
ronnie@fullcirclemagazine.org

ubuntu

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.

TIP: use the new 'contents' link to jump to the contents page from any other page!

Ubuntu 10.04 Beta Released

Ubuntu 10.04 LTS Desktop and Netbook Editions continue the trend of ever-faster boot speeds, with improved startup times and a streamlined, smoother boot experience. The Ubuntu 10.04 family of variants, Kubuntu, Xubuntu, Edubuntu, Ubuntu Studio, and Mythbuntu, also reach beta status today.

• On the Desktop: *GNOME 2.30, KDE SC 4.4, XFCE 4.6.1, OpenOffice.org 3.2.0, X.Org server 1.7.5*

• On the Server: *Apache 2.2, PostgreSQL 8.4, PHP 5.3.1, LTSP 5.2*

• "Under the hood": *GCC 4.4.3, eglibc 2.11, Linux 2.6.32.9, Python 2.6.5*

Full release notes and download links can be found at: http://www.ubuntu.com/testing/lucid/beta1

To upgrade from Ubuntu 9.10 or 8.04 LTS, follow these instructions: https://help.ubuntu.com/community/LucidUpgrades

The final version of Ubuntu 10.04 LTS is expected to be released in April 2010.

Source: Ubuntu Mailing List

Refreshing The Ubuntu Brand

The new style in Ubuntu 10.04 (below left, with the new Ubuntu logo) is inspired by the idea of "Light".

We're drawn to Light because it denotes both warmth and clarity, and intrigued by the idea that "light" is a good value in software. Good software is "light" in the sense that it uses your resources efficiently, runs quickly, and can easily be reshaped as needed. Ubuntu represents a break with the bloatware of proprietary operating systems [...]. More and more of our communications are powered by light, and in future, our processing power will depend on our ability to work with light, too.

Source: http://www.jonobacon.org/2010/03/03/refreshing-the-ubuntu-brand/

Full Circle Podcast - Episode 02 (& 03?)

That's right folks, the Full Circle Podcast is back and better than ever! The podcast is in both MP3 and OGG formats. Runtime is 48 mins.

By the time you read this, episode three should be hitting the internet.

Your Hosts:

- *Robin Catling*
- *Ed Hewitt*
- *Dave Wilkins*

The podcast and show notes are available at: http://fullcirclemagazine.org

AUDIO MP3

AUDIO OGG

COMMAND & CONQUER

Written by Lucas Westermann

After finishing my Screen segments, I realized that it may be interesting for my readers to see what other things the bash or zsh shells can do. Therefore, I'll be covering the various shells that exist for Linux (along with a short description), and an in-depth section on customizing/configuring *Z Shell* (Zsh) and *Bourne Again Shell* (Bash), since those are the two shells I've seen most widely used, and the two shells I'm most comfortable with. It also leaves room for you, my readers, to play around with a few new shells on your own and to learn for yourselves what they can do.

The following shells are available:

Bourne Shell (sh) – Original Unix shell. Offered no notable features outside of what one would expect from a shell.

Almquist Shell (ash) – BSD-Licensed re-write of Bourne Shell. Similar feature set as above.

Bourne-Again Shell (bash) – Standard shell used in Linux distributions. Offers a superset of Bourne Shell functionality. Written as part of the GNU Project.

Debian Almquist Shell (dash) – Modern replacement of the Almquist shell for Debian-based Linux distributions.

Korn Shell (ksh) – A shell written by David Korn.

Z Shell (zsh) – Considered the most “complete” shell available (offers the most features). Could be described as a superset of sh, ash, bash, csh, ksh, and tcsh (TENEX C shell).

C Shell (csh) – A shell written by Bill Joy, and is special in the sense that its syntax is similar to the c programming language.

This is by no means an exhaustive list of shells, but they're the ones I believe are still actively developed/used among the community. You may wonder why anyone would bother to change their default shell. The main reason why I prefer Z Shell over Bash is simply because it offers certain features I prefer (a better tab auto-completion than Bash, easier colour syntax for prompts, a right-hand prompt as well as a left-hand one, etc.). As with so many things to do with Linux, it ultimately boils down to choice. Maybe you're a skilled c programmer, and prefer to have a shell that has a similar syntax, and have therefore opted for the C Shell. I won't say one is better than another, simply because it has a feature or two that others don't, and vice versa. I will, however, only cover how to configure Bourne-Again Shell and Z Shell in this article, because I have experience with these, and because they seem to be the most widely used shells out there.

The first thing I need to cover is how to install and test a new shell, and how to change the default shell. To install, you just need to apt-get whatever shell you'd like to try out. Once it's installed, checking the manpage will give you the location of the configuration file. Also, since you'll most likely want to see the default prompt, you can switch shells by simply running the binary name for the shell. (sh, ash, bash, zsh, csh, ksh, and so forth). It will drop you into that shell without changing the default. I always recommend viewing the default configuration file, and making a local version for customization, in case something goes awry. I also recommend testing a new prompt via the command line, before committing it to the

configuration file. This is as simple as exporting the PS1 via the command-line. Just keep tweaking it until you're happy, and then copy the end product into the configuration page. Once you're happy with the configuration, and certain that there are no major problems with the configuration file, you are ready to change the default shell (as long as you want to). To do so, run the following command:

```
sudo chsh -s /path/to/binary $USER
```

You need to replace "/path/to/binary" with the path to the shell (i.e. /bin/bash), and $USER with your actual username/username of the account whose shell you'd like to change. In case you're not sure what shells you have available (and recognized by the system), you can view them using:

```
chsh -l
```

This may not show all the shells, since it merely prints those listed in /etc/shells, but most packages should update that file accordingly.

You may be wondering what exactly you can configure in a shell, and why you would bother. A few things that I will cover are: exporting environment variables for use in window managers (openbox instead of Gnome, for example), aliasing commands for easier use, customizing the prompt itself, and adding functions to the shell.

Configuring Z Shell

Follow this link: http://lswest.pastebin.com/WBm22Wig to view a full .zshrc file. A note on the bindkeys: this is due to the fact that Zsh lacks support for home/end/page up/page down, and displays only escape sequences when pressed, unless you define bindkeys such as I have. You may need to find the right escape sequence for the key. If you have vi emulation enabled (which is what I use, and is done with bindkeys -v), you can see the escape sequence by hitting ctrl + v, and then the key you wish to have the escape sequence for. Chances are the ones I use will work for most others though, so you can always try them first.

I'll be referring to the file for examples (using the line numbers as viewed on pastebin).

The first thing I'd like to cover is how to export variables, since it's a useful thing to know, and pretty easy to do. To export a variable, all you need to do is use the syntax:

```
export $VARIABLE="value"
```

As you can see in my configuration file on lines: 11, 15, 117, 82, 116, 131 and 132. You need to, of course, replace "$VARIABLE" with the actual variable (such as DE, or OOO_FORCE_DESKTOP), and "value" with the actual value. You can put quotes around the value, or leave them away if it's only one word (as you can see in the file). The last two exports in my configuration file are extremely useful when using openbox, since it sets the Desktop Environment to Gnome for xdg-open (the program that auto-selects the default application for filetypes). In other words "xdg-open" and a file path will open nautilus when set to Gnome, thunar when set to xfce, and konqueror when set to kde. The OOO_FORCE_DESKTOP export also sets OpenOffice to use the gtk theme, instead of using the QT theme, which is the default option unless the DE is gnome.

The next useful trick is to add aliases to your configuration file, so that you can use extended arguments for a command easily. This can be seen particularly well on line 84, since I use the alias trayer (thereby ignoring the actual binary file of that name), and use it to launch trayer with a specific set of arguments. If, however, you discover you want to use the original trayer binary without the alias, you can override the alias temporarily by using the following:

```
\<alias name>
```

So, in this case it would read \trayer. It's similar to how you escape certain characters so that a shell sees it only as text. An extremely useful alias I use on all my *nix boxes, is the alias for ls (on line 64), since it gives me a much more detailed listing of files within the folder.

Now we come to the most widely use customization of Shells. The prompt itself. The prompt I prefer to use in Z Shell is in the following format:

```
┌[lswest@laptop:~] - [14:24:29]
└>
```

It's a double-line prompt, giving me more room to write commands, and it offers me the current user logged in, the hostname, and the current working directory (after the colon). For those wondering how I manage a double-line prompt, the magic happens here: "$'\n'", where I break the main section of the prompt, and add in an escape sequence for a new line, and then continue the prompt. This doesn't work (last time I tried), with just the escape sequence in double quotes. Also, a right-handed prompt can be added using the RPROMPT variable (I have it commented out in my configuration file, but it is still there).

I update the configuration file regularly, and the copy that's on pastebin at the moment is an iteration or two behind, but the major change is that my current prompt also offers me the time I ran a command. If you look at the configuration file, you can see that there are actually two prompts listed in an if statement. Basically it checks to see if I'm using screen, and if so, it displays the current screen window value before my username, making it easy to keep track of where I am. A complete list of escape sequences for Zsh is available on the man page for zshmisc, but here is a list of ones I often use (taken from http://www.acm.uiuc.edu/workshops/zsh/prompt/escapes.html):

Literal characters

%% - A %
%) - A)

Directories

%d - The current directory ($PWD)

%~ - $PWD, but will do two types of substitutions. If a named dir 'X' is a prefix of the current directory, then ~X is displayed. If the current directory is your home directory, $HOME, just ~ is displayed.

%c - Trailing component of $PWD. If you want n tailing componenets, put an interger 'n' after the %.

%C - Just like %c and %. except that ~'s are never displayed in place of directory names.

Hostname info

%M - The full machine hostname.

%m - The hostname up to the first . (dot). An integer may follow the % to specify how many components of the hostname are desired.

Current time info

%t - Current time of day, in 12-hour, am/pm format.

%T - Current time of day, in 24-hour format.

%* - Current time of day in 24-hour format, with seconds.

Current date info

%w - The date in day-dd format.

%W - The date in mm/dd/yy format.

%D - The date in yy-mm-dd format.

%D{string} - string is formatted using the strftime function. See strftime(3) for more details. Three additional codes are available: %f prints the day of the month, like %e but without any preceding space if the day is a single digit, and %K/%L correspond to %k/%l for the hour of the day (24/12 hour clock) in the same way.

Miscellaneous info

%h - Current history event number.

%n - Equivalent to $USERNAME.

%l - The line (tty) the user is logged in on.

%# - A `#' if the shell is running with privileges, a `%' if not. The definition of privileged, for these purposes, is that either the effective user ID is zero, or, if POSIX.1e capabilities are supported, that at least one capability is raised in either the Effective or Inheritable capability vectors.

Zsh offers a few default colours that can be accessed with names such as red, cyan, etc. But it also accepts the usual \e[0;31m style formatting (as discussed in the Bash section).

Last, but possibly most useful, is the ability to add functions to a shell. The set up is exactly the same as for Bash scripts. The method is to define a function with “function name() { #code }”. I have a few functions in my Zshrc file, such as: m4a, flvmp3, google, etc. As you can see, you can also define a function without using the descriptor “function”, but it makes it more readable. My configuration file is by no means a good example of an organized file. Ideally, I'd have kept all exports together, all functions together, all aliases, and so forth. Instead, I add things to the file as I think of them, leaving me with a bit of a mess. I'll probably get around to tidying it up eventually (seems to happen about once a year)

Configuring Bourne-Again Shell

Exporting and aliasing are exactly the same for Bash shells as for Zsh shells, so to see how to do that, please read the first two explanations of the Configuring Z Shell section. The only sections in the .zshrc file I have a link to that aren't relevant to Bash prompts are the bindkeys section, and the PROMPT sections.

As for customizing the prompt in Bash, it's similar to Zsh, except for the list of escape sequences you can use, and how the variable behaves when it comes to double-lines. The following is a list of escape sequences for bash (taken from: http://www.cyberciti.biz/tips/howto-linux-unix-bash-shell-setup-prompt.html):

\a : an ASCII bell character (07)

\d : the date in "Weekday Month Date" format (e.g., "Tue May 26")

\D{format} : the format is passed to strftime(3) and the result is inserted into the prompt string; an empty format results in a locale-specific time representation. The braces are required

\e : an ASCII escape character (033)

\h : the hostname up to the first '.'

\H : the hostname

\j : the number of jobs currently managed by the shell

\l : the basename of the shell’s terminal device name

\n : newline

\r : carriage return

\s : the name of the shell, the basename of $0 (the portion following the final slash)

\t : the current time in 24-hour HH:MM:SS format

\T : the current time in 12-hour HH:MM:SS format

\@ : the current time in 12-hour am/pm format

\A : the current time in 24-hour HH:MM format

\u : the username of the current user

\v : the version of bash (e.g., 2.00)

\V : the release of bash, version + patch level (e.g., 2.00.0)

\w : the current working directory, with $HOME abbreviated with a tilde

\W : the basename of the current working directory, with $HOME abbreviated with a tilde

\! : the history number of this command

\# : the command number of this command

\$: if the effective UID is 0, a #, otherwise a $

\nnn : the character corresponding to the octal number nnn

\\ : a backslash

\[: begin a sequence of non-printing characters, which could be used to embed a terminal control sequence into the prompt

\] : end a sequence of non-printing characters

To make a multi-line prompt in Bash, all you need to do is place an escape sequence newline character ("\n") where you'd like the line to break. You can also customize PS2 and onwards, which appear when you start a multi-line command (e.g. A for loop). As for colors, the escape sequences are available (from: http://wiki.archlinux.org/index.php/Color_Bash_Prompt#List_of_colors_for_prompt_and_Bash):

You can, of course, place the colors inside variables and use that within the configuration file. The bash version of my Zsh prompt (without the timestamp) would be as follows:

```
export PS1="\[\e[0;37m\]┌─[\[\e[0;32m\]\u\[\e[0;36m\]@\[\e[0;32m\]\h\[\e[0;37m\]:\[\e[0;33m\]\w\[\e[0;37m\]]\[\e[0;36m\]\n\[\e[0;37m\]└─>\[\e[0m\] "
```

I apologize for not having an example bash file to display, but the configuration syntax for both Zsh and Bash are similar, so that should be a decent example for both. If any reader would like, I would be happy to display your customized .bashrc files, along with a textual representation of the prompt, at the beginning of each month's Command & Conquer. If you're interested, just send me an email at lswest34@gmail.com with your .bashrc, and a textual representation of the prompt, or an actual screenshot. Also, please refer to Command & Conquer in the subject line, so that I put it higher on my priority list. For any users who use urxvt/define custom prompt colors in your .Xdefaults, please share the relevant section as well (if you send a screenshot).

Any questions, suggestions, or problems can be emailed to me at lswest34@gmail.com, and any further ideas for segments are always welcome in my inbox! I wish you all a fun time configuring your prompts, and I'm curious to see what your results will be! I hope I've done a good job at explaining this, and I'll gladly continue on with further customizations to the terminal, if there's enough interest. And, as always, there is plenty more information regarding this in the Further Reading section.

Further Reading

http://en.wikipedia.org/wiki/Alias_%28command%29 – Info on the Alias command

http://www.cyberciti.biz/tips/howto-linux-unix-bash-shell-setup-prompt.html – Bash prompt customization how-to

http://markelikalderon.com/2007/11/24/full-paths-and-the-multiline-shell-prompt/ - Multi-line prompts.

http://wiki.archlinux.org/index.php/Color_Bash_Prompt#List_of_colors_for_prompt_and_Bash – colorizing bash prompts

http://docs.cs.byu.edu/linux/advanced/zsh.html – how to configure Zsh prompts.

Lucas has learned all he knows from repeatedly breaking his system, then having no other option but to discover how to fix it. You can email Lucas at: lswest34@gmail.com.

Program In Python - Part 9

Written by Greg Walters

SEE ALSO:
FCM#27-34 - Python Parts 1 - 8

APPLICABLE TO:

ubuntu kubuntu xubuntu

CATEGORIES:

Dev Graphics Internet M/media System

DEVICES:

CD/DVD HDD USB Drive Laptop Wireless

If you are anything like me, you have some of your favorite music on your computer in the form of MP3 files. When you have less than 1000 music files, it's rather easy to remember what you have and where it is. I, on the other hand, have many more than that. In a past life, I was a DJ and converted most of my music a number of years ago. The biggest problem that I had was disk space. Now the biggest problem is remembering what I have and where it is.

In this and the next installment we will look at making a catalog for our MP3 files. We will also take a look at some new python concepts as well as re-visiting our database skills.

First, an MP3 file can hold information about the file itself. The title of the song, the album, artist and more information. This information is held in ID3 tags and is referred to as metadata. Back in the early days, there was only a limited amount of information that could be held inside of the MP3 file. Originally, it was stored at the very end of the file in a block of 128 bytes. Because of the small size of this block, you could only hold 30 characters for the title of the song, name of the artist, and so on. For many music files, this was fine, but (and this is one of my favorite songs ever) when you had a song with the name "Clowns (The Demise of the European Circus with No Thanks to Fellini)", you only got the first 30 characters. That was a BIG frustration for many people. So, the "standard" ID3 tag became known as ID3v1 and a new format was created called, amazingly enough, ID3v2. This new format allowed for variable length information and was placed at the beginning of the file, while the old ID3v1 metadata was still stuck at the end of the file for the benefit of the older players. Now the metadata container could hold up to 256 MB of data. This was ideal for radio stations and crazies like me. Under ID3v2, each group of information is held in what's called a frame and each frame has a frame identifier. In an earlier version of ID3v2, the identifier was three characters long. The current version (ID3v2.4) uses a four character identifier.

In the early days, we would open the file in binary mode, and dig around getting the information as we needed it, but that was a lot of work, because there were no standard libraries available to handle it. Now we have a number of libraries that handle this for us. We will use one for our project called Mutagen. You will want to go into Synaptic and install python-mutagen. If you want, you could do a search for "ID3" in Synaptic. You'll find there are over 90 packages (in Karmic), and if you type "Python" in the quick search box, you'll find 8 packages. There are pros and cons with any of them, but for our project, we'll stick with Mutagen. Feel free to dig into some of the other ones for your extended learning.

Now that you have Mutagen installed, we'll start our coding.

Start a new project and name it "mCat". We'll start by doing our imports.

```
from mutagen.mp3 import MP3

import os

from os.path import
join,getsize,exists

import sys

import apsw
```

For the most part, you've seen these before. Next, we want to create our stubbed function headers.

```
def MakeDataBase():
     pass
def S2HMS(t):
    pass
def WalkThePath(musicpath):
     pass
def error(message):
    pass
def main():
    pass
def usage():
    pass
```

Ahhh...something new. We now have a main function and a usage function. What are these for? Let's put one more thing in before we discuss them.

```
if __name__ == '__main__':
    main()
```

What the heck is that? This is a trick that allows our file to be used as either a stand alone application or a re-usable module that gets imported into another app. Basically it says "IF this file is the main app, we should go into the main routine to run, otherwise we are going to use this as a utility module and the functions will be called directly from another program.

Next, we'll flesh out the usage function. Below is the full code for the usage routine.

Here we are going to create a message to display to the user if they don't start our application with a parameter that we need to be able to run as a standalone app. Notice we use '\n' to force a new line and '\t' to force a tab. We also use a '%s' to include the application name which is held in the sys.argv[0]. We then use the error routine to output the message, then exit the application (sys.exit(1)).

Next, let's flesh out the error routine. Here is the full error routine.

```
def error(message):
    print >> sys.stderr,
str(message)
```

We are using something called redirection here (the ">>"). When we use the function "print", we are telling python we want to output, or stream, to the standard output device, usually the terminal that we are running in. To do this we use (invisibly) stdout. When we want to send an error message, we use the stderr stream. This is also the terminal. So we redirect the print output to the stderr stream.

Now, let's work on the main routine. Here we will setup our connection and cursor for our database, then look at our system argument parameters, and if everything is good, we'll call our functions to do the actual work we want done. Here's the code:

```
def usage():
    message = (
       '================================================\n'
       'mCat - Finds all *.mp3 files in a given folder (and sub-folders),\n'
       '\tread the id3 tags, and write that information to a SQLite database.\n\n'
       'Usage:\n'
       '\t{0} <foldername>\n'
       '\t WHERE <foldername> is the path to your MP3 files.\n\n'
       'Author: Greg Walters\n'
       'For Full Circle Magazine\n'
       '================================================\n'
       ).format(sys.argv[0])
    error(message)
    sys.exit(1)
```

```
def main():
    global connection
    global cursor
    #-------------------------------------------
    if len(sys.argv) != 2:
        usage()
    else:
        StartFolder = sys.argv[1]
        if not exists(StartFolder): # From os.path
            print('Path {0} does not seem to
exist...Exiting.').format(StartFolder)
            sys.exit(1)
        else:
            print('About to work {0}
folder(s):').format(StartFolder)
        # Create the connection and cursor.
        connection=apsw.Connection("mCat.db3")
        cursor=connection.cursor()
        # Make the database if it doesn't exist...
        MakeDataBase()
        # Do the actual work...
        WalkThePath(StartFolder)
        # Close the cursor and connection...
        cursor.close()
        connection.close()
        # Let us know we are finished...
        print("FINISHED!")
```

As we did last time, we create two global variables called connection and cursor for our database. Next we look at the parameters (if any) passed from the command line in the terminal. We do this with the sys.argv command. Here we are looking for two parameters, first the application name which is automatic and secondly the path to our MP3 files. If we don't see two parameters, we jump to the usage routine, which prints our message to the screen and exits. If we do, we fall into the else clause of our IF statement. Next, we put the parameter for the starting path into the StartFolder variable. Understand that if you have a path with a space in it, for example, (/mnt/musicmain/Adult Contemporary), the characters after the space will be seen as another parameter. So, whenever you use a path with a space, make sure you quote it. We then setup our connection and cursor, create the database, then do the actual hard work in the WalkThePath routine and finally close our cursor and connection to the database and then tell the user we are done. The full WalkThePath routine can be found at: http://pastebin.com/CegsAXjW.

First we clear the three counters we will be using to keep track of the work that has been done. Next we open a file to hold our error log just in case we have any problems. Next we do a recursive walk down the path provided by the user. Basically, we start at the provided file path and "walk" in and out of any sub-folders that happen to be there, looking for any files that have a ".mp3" extension. Next we increment the folder counter then the file counter to keep track of how many files we've dealt with. Next we we step through each of the files. We clear the local variables that hold the information about each song. We use the join function from os.path to create a proper path and filename so we can tell mutagen where to find the file. Now we pass the filename to the MP3 class getting back an instance of "audio". Next we get all the ID3 tags this file contains and then step through that list checking for the tags we want to deal with and assigning them to our temporary variables. This way, we can keep errors to a minimum. Take a look at the portion of code dealing with the track number. When mutagen returns a track number it can be a single value, a value like "4/18" or as _trk[0] and _trk[1] or it can be absolutely nothing. We use the try/except wrappers to catch any errors that will occur due to this. Next, look at the writing of the data records. We are doing things a bit different from last time. Here we create the SQL statement like before, but this time we are replacing the value variables with "?".

We then put in the values in the cursor.execute statement. According to the ASPW web site, this is the better way to deal with it, so I won't argue with them. Finally we deal with any other types of errors we come up with. For the most part, these will be TypeErrors or ValueErrors and will probably occur because of Unicode characters that can't be handled. Take a quick look at the strange way we are formatting and outputting the string. We aren't using the '%' substitution character. We are using a "{0}" type substitution, which is part of the Python 3.x specification. The basic form is:

```
Print('String that will be printed with {0} number of statements").format(replacement values)
```

We are using the basic syntax for the efile.writelines as well.

Finally we should take a look at the S2HMS routine. This routine will take the length of the song which is a floating point value returned by mutagen and convert it to a string using either "Hour:Minutes:Seconds" format or "Minutes:Seconds" format. Look at the return statements. Once again, we are using the Python 3.x formatting syntax. However, there's something new in the mix. We are using three substitution sets (0, 1 and 2), but what's the ":02n" after numbers 1 and 2? That says that we want leading zeros to two places. So if a song is 2 minutes and 4 seconds, the returned string would be "2:04", not "2:4".

The full code of our program is at: http://pastebin.com/rFf4Gm7E.

Dig around on the web and see what you can find about Mutagen. It does more than just MP3s.

Greg Walters is owner of *RainyDay Solutions, LLC*, a consulting company in Aurora, Colorado, and has been programming since 1972. He enjoys cooking, hiking, music, and spending time with his family.

MY STORY QUICKIE

My studio is fully digital with four Windows XP machines in a peer to peer network. My fifth machine runs Linux Ubuntu 9.04 exclusively as my test machine for Linux. I started with Ubuntu 7.04 and have upgraded each time there was a release. I have found it to be very stable, easy to use and configure as each version improves the OS.

At this time it is only my test bed but is linked to my network and shares data with my Windows machines. I have been very happy with the stability of Ubuntu in its upgrades, programs, hardware support, and driver updates. Although it is unfortunate that more major vendors such as Adobe don't port over, but Wine seems to work well. There are graphics programs and professional printers related to my camera equipment that do not work so I will have to wait until Wine gets better or the software gets ported over.

Audio, video, CD/DVD, USB, and Zip drives all seem to work 'out of the box' which is nice. Still some flaws in the software but they appear to be minor annoyances.

All in all Ubuntu has been visually refreshing and fun to play with. I am not a geek so I really do not use the command line unless curious about a tutorial and want to try it, the OS GUI is quite complete for us non-geeks who want to stick to a GUI.

I download Full Circle Magazine every month and have shared it with one of my colleagues to show him what is available. A lot of people still do not know about the OS and how easy it is to use, but as the Microsoft disgruntled get the word out I expect to see more growth. The one thing I absolutely love about this OS is the ability to shut down a misbehaving program. The break button works slickly in Linux and eliminates the frustration of waiting for Windows to unfreeze in XP. Why can't Windows do something as easy as that? I seldom need to use the button in Linux anyway which shows how stable Linux is.

Brian G Hartnell - *Photographer*

HOW-TO

Written by Hüseyin SARIGÜL

Retouch Photos in GIMP - Part 2

SEE ALSO:
FCM#34 - Retouch Photo's - Part 1

APPLICABLE TO:

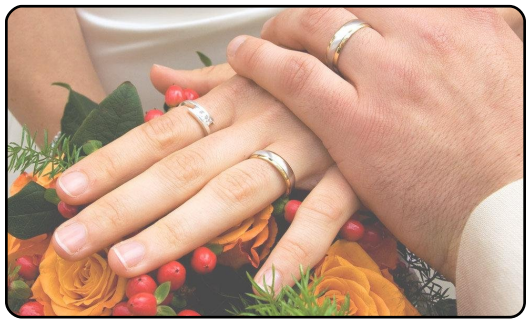
ubuntu kubuntu xubuntu

CATEGORIES:

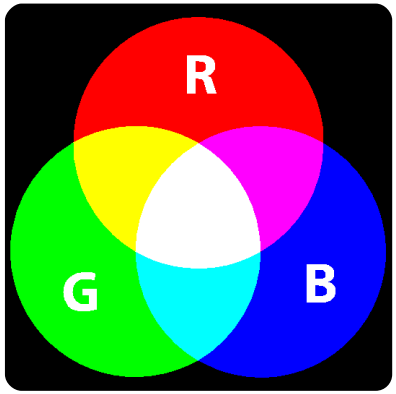
Dev Graphics Internet M/media System

DEVICES:

CD/DVD HDD USB Drive Laptop Wireless



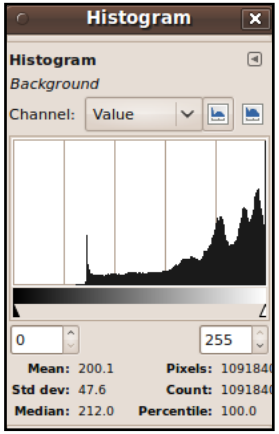
In this article we will learn how to edit dark and light tones in our photos. First of all, I would like explain colors and their ranges. There are different color profiles, we have three main colors within light and they are Red, Green and Blue (RGB). Another color profile is CMYK this is a mix of the main ink colors and is usually used in photos, printers and press.



Histogram

The Histogram graph shows us the range of all the colors or a single color. How should be the color graph look? Let's explain it with some examples.

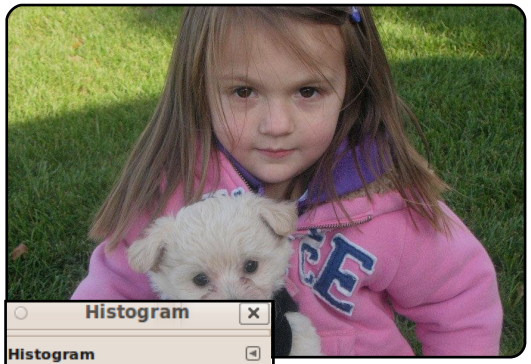
Above we have a photo with light and mid tones but no dark tone.

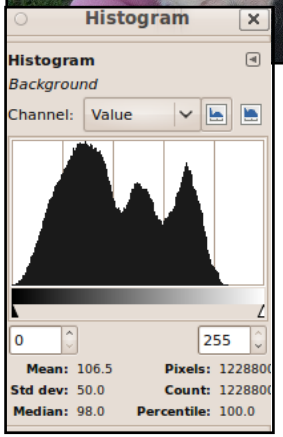


Now we go to Dockable Dialogs > Histogram from the menu. If you check the histogram (left), you can easily see which tones are not in your photo.

Let's check our second example (above right).

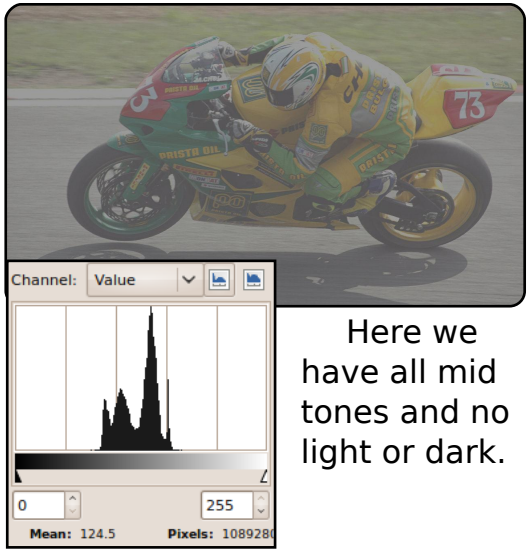
In this histogram, we have





mid and dark tones but no light tones (right).

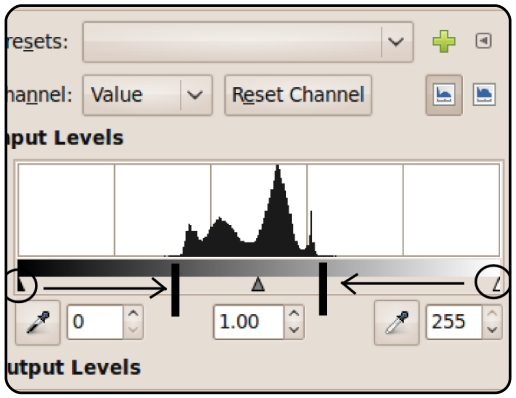
Now, our third example:



Here we have all mid tones and no light or dark.

If there is a problem with the colors in a photo, we can easily correct things by using either the Levels dialog or Brightness/Contrast settings.

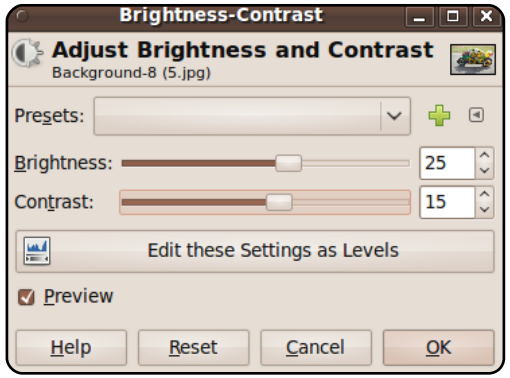
Now we choose Colors > Levels. This pop-up represents the mix of RGB colors as it stands.



From “Channel” you can reach the histogram of RGB. We can edit them one by one. The “Auto” button can edit the tones automatically, but this isn't always correct.

There are three “Pick” buttons next to the "Auto” button. They are for dark , mid and light tones. You can click them one by one and choose the suitable tone point.

The best way is by doing it manually. You can do this by dragging the triangles, which I have marked with circles (left), to the start and end points of the histogram. Then you can find the best tone by changing the position of the middle triangle. You can then use Brightness/Contrast to make colors sharp.



Let's go to Colors > Brightness/Contrast and choose:
+25 Brightness
+15 Contrast

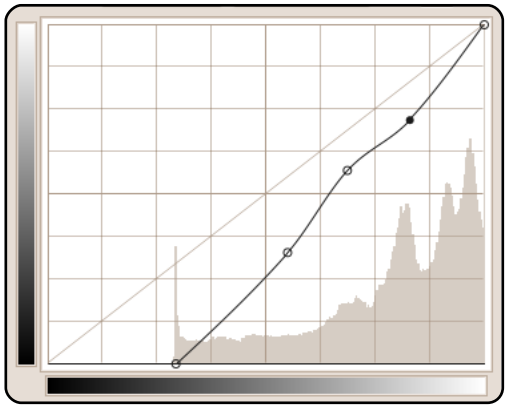
Now our photo should look like this.

You can edit the tone of the other photos as we did there.

The second tool we can use is the Adjust Color Curves. This tool has similar properties as the “Levels” tool but it has a few little extras. With it we can increase the density of colours.

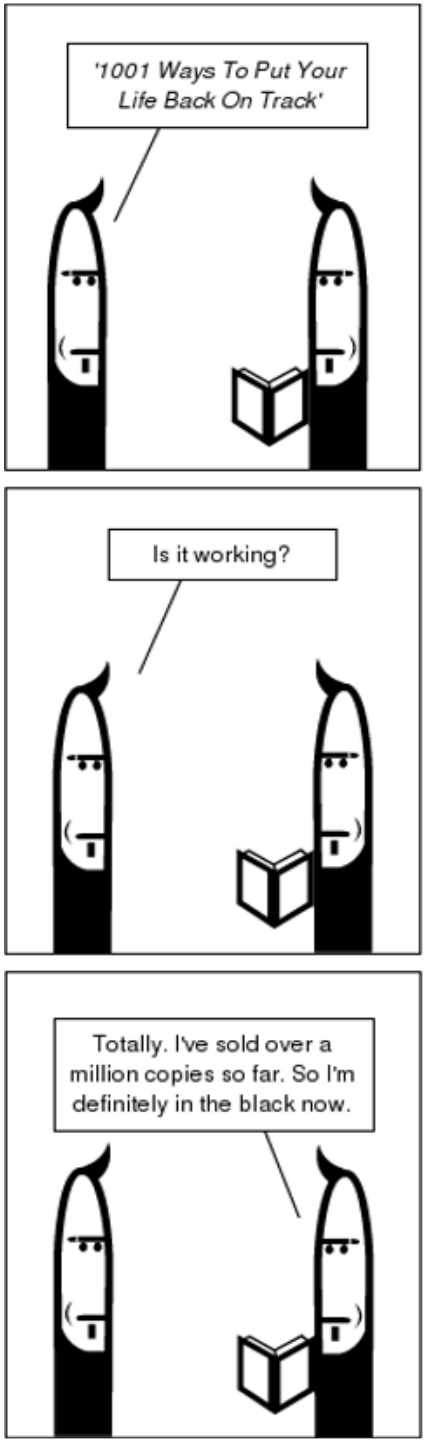
Let's open our first photo and choose Colors > Adjust Color Curves. In the pop-up window, the starting and ending points can be changed and tones can be adjusted.

That's all for this issue. The next article will tell you more about colour adjustment.



All photos are licensed by Public Domain.

Translated from Turkish to English by Kaan Bahadır TERMELİ



HOW-TO
Written by Ronnie Tucker

Install Google SketchUp Using Wine

SEE ALSO:
N/A

APPLICABLE TO:
ubuntu kubuntu xubuntu

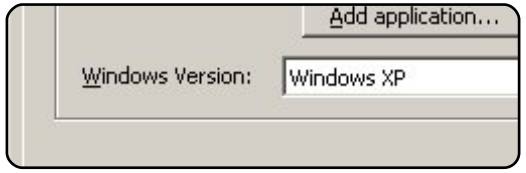
CATEGORIES:
Dev Graphics Internet M/media System

DEVICES:
CD/DVD HDD USB Drive Laptop Wireless

Google SketchUp is a very popular, and free, CAD style application. So far so good, but the bad news is that it is written for Windows. The good news is that it will run under Wine with very little difficulty these days.

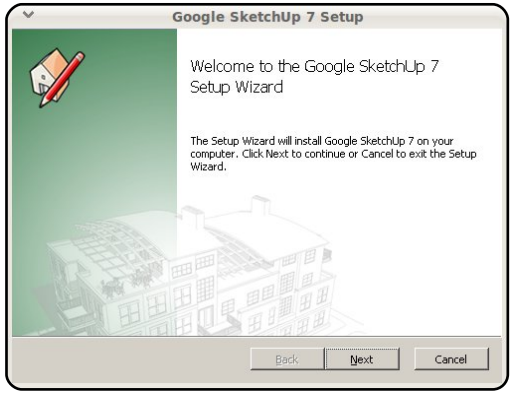
The first thing you must do is, of course, download SketchUp from: http://sketchup.google.com/intl/en/download/gsu.html. Next, you need to either install the latest version of Wine, or update your current installation of Wine to the most recent version. I recommend you use the Wine (Ubuntu) repo's, how to add the repository is at: http://www.winehq.org/download/deb. To check your version of Wine, click Applications > Wine > Configure Wine, and click the 'About' tab. To get SketchUp working properly, it's best to have at least 1.1.11 of Wine

While you have the configuration window open, click the 'Applications' tab, and set the default Windows to XP



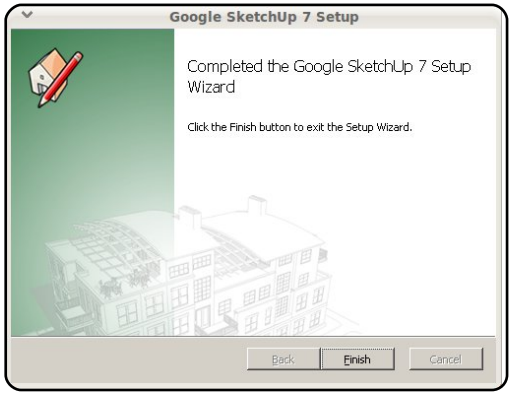
The last thing you should do is to make sure you have nVidia drivers installed (if need be), the Wine homepage says: '*you probably need nvidia's proprietary graphics driver. The open-source nv driver isn't good enough yet*'.

With all the necessary steps in place, it's time to install SketchUp! Open the folder of where you saved the SketchUp .exe file, right click on the ***GoogleSketchUpWEN.exe*** file and choose 'Open with Wine Windows Program Loader'.



Don't double click it though, as Ubuntu will think it's an archive and just display its contents.

Simply follow the prompts, accepting all the default settings, and it should install no problem at all.



The installer will put two links on your desktop but, since those are for Windows, you can delete them.

We're not done yet. Google SketchUp needs some Windows files, and the easiest way to install them is using a script called winetricks. To install wintricks, load up a command line and issue the command:

```
wget http://www.kegel.com/wine/winetricks
```

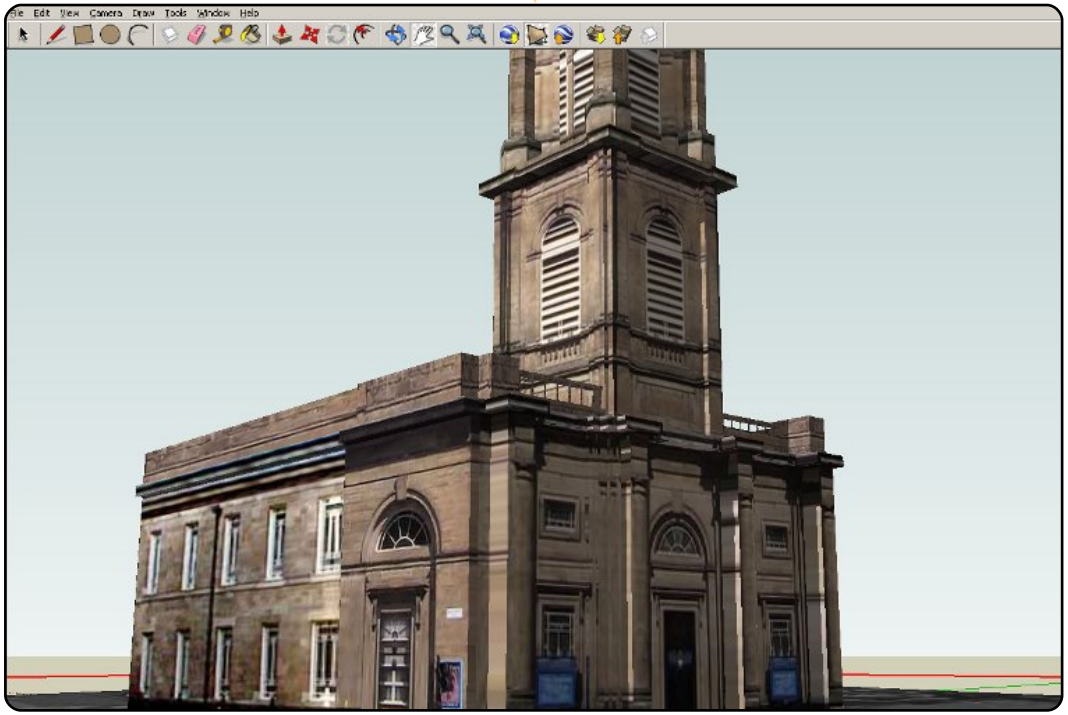
then to install the actual Windows files:

```
sh winetricks corefonts vcrun6 vcrun2005
```

NOTE: Do not install any .dll's or Windows system files manually as this will cause a ton of errors, I know... I did it the first time I installed SketchUp!

Once winetricks is done installing the Windows files, you click Applications > Wine > Programs > Google SketchUp 7 > Google SketchUp and SketchUp will load.

The first time I tried installing SketchUp I couldn't get it to load, but deleting my .wine directory and reinstalling SketchUp made it work no problem at all, but beware: deleting the .wine directory will delete everything you have on your Wine C: drive!



Full Circle Podcast

AUDIO MP3 AUDIO OGG

The ***Full Circle Podcast*** is back and better than ever!

Topics this episode #2 include:
- Karmic 64-bit
- Lucid Lynx - Alpha 3
- Ubuntu One Music Store
- Sauerbraten

and Chuck Norris attacks Linux routers.

Your Hosts:
- *Robin Catling*
- *Ed Hewitt*
- *Dave Wilkins*

The podcast and show notes are at:
http://fullcirclemagazine.org/

My Introduction To Ubuntu

Written by Rahul Mehta

Windows was giving me trouble, a lot of trouble. Keeping my PCs running was getting to be a job by itself, taking valuable time. I was looking for something that could take the place of Windows. I had two simple criteria: it should be sufficient to do my work with documents, pictures and sound, and be easy to use. I considered Linux but did not want to go back to command line operations again. Mac OS was out as it won't run on PCs.

Then I came across an article in a magazine that talked about Live CD Linux. I burned one and tried it out the same day. Live CD worked pretty fine. A bit slow, but it gave me a fairly good idea of how it would be. At first I couldn't get things to work. But slowly, order emerged from chaos. I sort of liked it but was not greatly impressed. My conclusion was that it would take a lot of time to learn the new system. It was not friendly enough for me. That was that. I went back to Windows.

But the need kept pulling me back to Linux. I searched and found several different distros offering live CD versions. I started trying them out one after another. That is when I came across Ubuntu – Feisty Fawn. I liked it first time. The easy-to-use windowing system gave me the idea that this could be “the one”. However, I kept trying others including Kubuntu. Then came Gutsy Gibbon. That clinched the deal. I found Ubuntu's Gnome easier to use than Kubuntu's KDE.

I started transferring my work from Windows to Ubuntu. A lot of hiccups and stalls but, as I was getting used to the system, it became easier and easier. On PCs under Windows, I already was using open software to do my tasks – mainly OpenOffice, GIMP and Audacity among other programs. So transferring those files was pretty easy. It was the task of managing the computer that was taking time to learn, for example, setting up the PCs on the network. But as I understood the system better, it became easier to use.

I must have installed and reinstalled the OS a dozen times, if not more, just to make it work. The help I got from numerous on-line sources was invaluable. It still is. If I had problem, I would search for the symptoms, or the error, and could find a cure in a short time. I must admit though, using terminal and command line operations is not new for me. I come from the Pre-DOS era of computers. Even in Windows I still am comfortable using DOS commands. However, the idea of remembering new commands was not very comforting. For me, Ubuntu has, by far, the best UI I have seen.

When Hardy Heron was released, I downloaded and installed it the day it was available. My laptop was the first to get Hardy Heron then the desktop. Once the network was up, I was on my way to the Linux world. I still have XP and Vista machines that my family use, but I am all Ubuntu. In fact, the mix of the OS gave me a chance to make it work in the heterogeneous network environment, an experience in itself.

I still am in a learning stage with Linux and am enjoying the experience. My confidence in Ubuntu is increasing day by day but it is still giving me problems in some areas, hence I still have Windows in the other partition that I hardly visit - once a month to keep it updated and occasionally for work.

On the whole, Linux and Ubuntu are in my future plans. My commitment is shown by my refusal to upgrade to Vista and now to Windows 7.

MY STORY

Written by Sean Rallis

How I Discovered Ubuntu

I stumbled upon Full Circle magazine because of a whim. I have been using Ubuntu for about 3 years, and have loved every second of it, so I'm always trying to find ways to get involved. Well, yesterday, while I was being an internet rat, I said to myself, “I wonder...” and typed “Ubuntu magazine” into my Google search. The first hit in the list was Full Circle magazine. It looked very intriguing, so I downloaded the latest issue and skimmed through it. I loved it! It was packed with current events, how-to's, and interviews pertaining to the world's best operating system: Linux. It filled my nerdy side with joy, and I downloaded all of the issues. I was up all night reading until my wife finally made me turn off the light at 3:00 am. So, in an effort to involve myself, I figured I'd write an article about my discovery and experience of Linux. I hope to have it put in a future issue, but if that doesn't happen, I'm still having fun.

People are always a little surprised to find that I'm a computer nerd. I usually don't talk about it unless someone else brings up the topic. But people come to my house, and the first thing that greets them is a 160 pound Great Dane. Then they go upstairs and find that my wife and I have 6 snakes, some of which are over six feet long. Most people also find out that I am a gun enthusiast, and work at a local shooting range. So when the topic comes up, and I start babbling about Linux and source code, and programming... I usually get more strange looks than the snakes!

My experience with Linux started when I was a cab driver in Burlington, a relatively small town in northeastern Vermont. I knew next to nothing about software, but I was good with tinkering, and was building a computer. I was using shoddy old parts that I got from friends who didn't need them, but I did manage to build a functional computer. The specs were horrible. I had a 6.4 GB hard drive, an ancient graphics card, no sound card, 128 MB of RAM, and a monitor that would sometimes shut off randomly. As far as I knew, the only operating systems in existence were Windows and Mac OS. Windows XP wouldn't have worked on this thing, so I ended up getting an old copy of Windows 98 Second Edition from my cousin. It wasn't much, but it worked. In a manner of speaking.

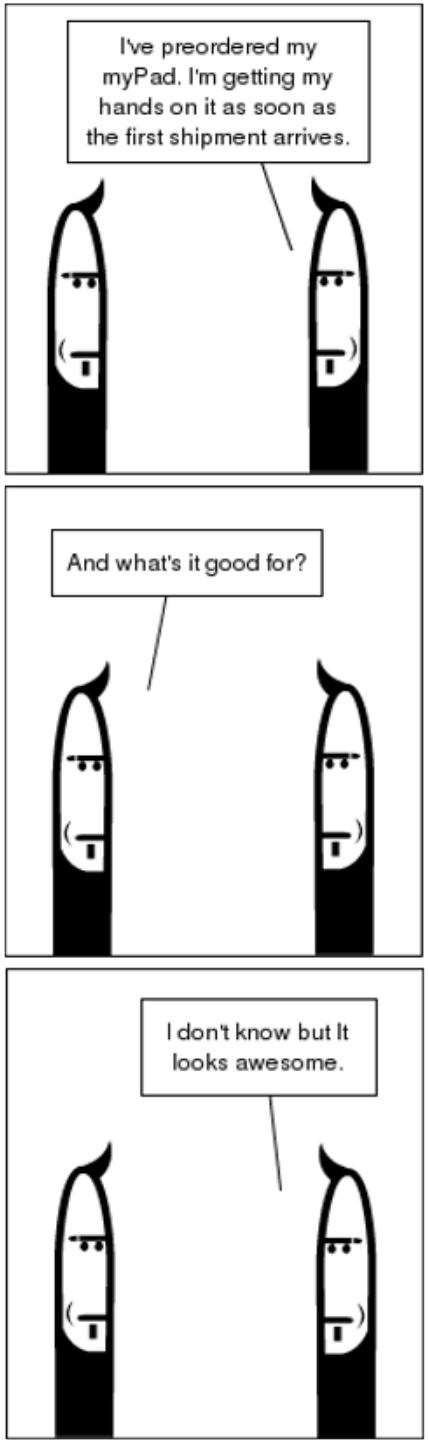
Did I mention the horrible graphics card? Yeah. Windows couldn't even use it properly, and wouldn't let me go beyond 16 colors, even though the card was capable of 256. I installed a USB port on it and bought a Linksys unit to connect to my roommate's wireless signal. Shockingly, it worked. I couldn't really see the images, but I could read text.

Then, I was at work one day telling another cabbie about my overgrown calculator. He asked me if I had tried putting Linux on it. I said, “What's Linux?” He said, “Linux is the operating system that's kicking Microsoft's butt.” So I went home after my shift and fired up my computer. I did some research on this “Linux” and found that Ubuntu, the most popular distro, wouldn't work on my computer. But Xubuntu would. I downloaded Xubuntu, but I didn't have a CD burner. I put the .iso on a flash drive, and used my roommate's computer to burn it to a CD. While installing, it gave me

the option of installing alongside Windows. I had read about dual-booting earlier and thought I would give it a try, just in case. Yes, you heard right. I was dual-booting with a 6.4 GB hard drive. 3GB for Windows, 3GB for Xubuntu, and the remainder I assigned to swap. After it installed, I rebooted and selected Xubuntu from the grub menu. Xubuntu started up, and a whole new world was presented to me. I was shocked at how well it worked! My graphics card was supported! I didn't have to restart for every piece of hardware! It all "**JUST WORKED!!!**" It required no input on my part, and just used all of my hardware as if it was designed for my specific computer. I explored a little bit, and looked at some of the software that came bundled with it. Then I plugged in my Linksys to connect to the internet and ran into my first Linux problem. It wouldn't work. I remembered seeing something called "Ubuntu Forums" in my research, so I rebooted to Windows and made an account with ubuntuforums.org and posted my question. Ten minutes later, I received an email saying someone had replied. I was expecting to get a response in a day at the least. The fact that someone responded that quickly was kind of shocking to me. I checked the response, and as it turned out, there were three. One of them was from a guy named *Jason*. Jason went far above and beyond simple suggestions. When his initial suggestions did not work, he made solving my problem his personal vendetta, and we discussed solutions via instant messaging. It was a pain, because I had to be on Windows to chat, then reboot to Linux to try solutions, then boot back to Windows to report. Well, long story short, we never did get the Linksys working. It turned out that it had a chip set specifically designed not to work under Linux. However, throughout this whole ordeal, Jason educated me about Linux, and some of the software available. He thus opened my eyes to the open source community.

The rest, as they say, is history. I finally got rid of that old junk computer, and have been upgrading my system whenever I can. I am now running Ubuntu 9.04 and will be downloading Karmic as soon as I get around to it. In short, my experience with Ubuntu has been fun. I've tried tons of distros, but so far have stayed with Ubuntu as my main OS. I even got my wife to start using it, and she loves it.

Is Ubuntu flawless? Certainly not. But is any operating system perfect? Of course not. But in my book, Ubuntu is as close as it gets.



I've seen recent arguments, pro and con, about making Linux an industry standard OS, that is, adding as many users as possible.

Depending upon who you talk to, Linux has less than 5% of the operating system market. We all know who's in first place with Apple, the last time I checked, chugging along with somewhere around 10%.

Is going mainstream a wise choice? For the purposes of this argument, I'm going to play the middle ground and present both sides and let the reader decide.

On a positive note, increasing Linux usage and awareness:

• **Might lead to additional funded research on making the OS versions more compatible with existing computer engineering.** One of the major complaints from potential users is that some computers can digest Linux without problems, yet others are nothing but headaches. Developers don't have much incentive to update systems as they should, unless they are somehow able to charge fees to offset costs (sort of like Canonical charging for tech support that in turn helps to support additional upgrades). This results in upgrades that are often geared more to the eye candy aspect of an OS than the underlying code and compatibility issues.

• **Might attract the interest of programmers and developers currently pegged to Microsoft or Apple products because, "that's where the money is".** Most college graduates in computer sciences are interested in the bucks, not what is best for society as a whole. Being a Linux developer for a small outfit doesn't pay as well as working for MS or Apple.

• **Could attract scholarly attention.** When I attended various state colleges in Florida not one of them discussed Linux let alone taught courses in the programming or protocol. It was considered the ugly stepchild and many graduates had to go to other schools to get the proper training once they were hired by corporations using Linux.

• **Can help end the crippling monopoly currently in place where Microsoft and Apple are the big boys and tough if you don't like it.** Where else but Microsoft can you pay $200 for an operating system and have it cancelled if you refuse to properly register it?

• **Could help end the secondary monopoly whereby Microsoft controls the word processing and office suite market.** Let's see, Corel and Lotus have all but bought the farm in this arena and how many other companies did Microsoft put out of business when they bought rights to their current stable of software? Anybody remember the original spreadsheet program called Jazz? You might know it better as Excel today. Why create something new when you can have somebody else do it and buy the rights?

• **May end the era of grossly overpriced computers.** Apple builds their computers and charges whatever the market will bear and, for whatever reason, has a loyal following willing to pay the tariff. Could it be their computers last years without dying? Microsoft doesn't build the computers but believe me when I say you do pay for them slapping a Microsoft OS in the mix. If Linux is offered in the retail environment will it make a difference in what the big boys charge?

• **Might be able to preserve the exclusive nature of the group without destroying**

it. Be truthful and admit that one of the reasons you use Linux is because it's exclusive. You and I are elitist computer snobs. Check your neighborhood and I doubt anyone else uses Linux. Same goes for Apple. Of 190 homes in my housing area, I'm the lone wolf using Apple and Linux. Even going mainstream doesn't mean you'll lose your exclusivity.

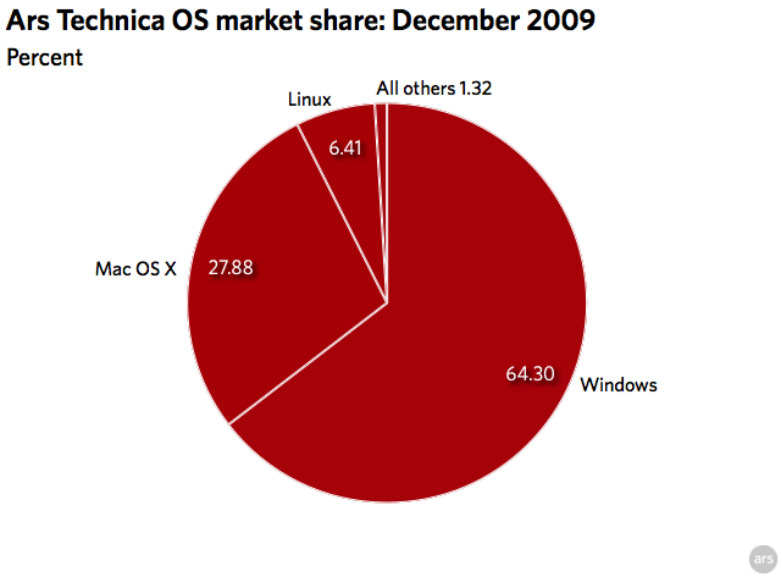
• **Could revitalize dying corporate interests currently pinned to other projects.** Corel and Lotus, among others, have all but been decimated by Microsoft and Apple. They could reinvent themselves by going over to the Linux family with a set of stable and cheap software geared to compete, as long as we don't remember the ill-fated Corel Linux from quite a few years ago.

• **Might get professors and users alike to pronounce the name right.** It's pronounced Lenox, not like it's spelled. Can't tell you how many teachers I've had to correct. And the major originator's first name is Linus, not Linux.

On the negative side, increasing usage and awareness:

• **Might give malcontents a reason to create viruses.** As it currently stands, Linux is a safe OS only because few virus creators will fool around with an OS that has less than 5% of the market. Apple found themselves in the same cart until recent years when their OS suddenly gained in popularity. Now they are stuck with issuing patches for potential security issues just like Microsoft.

• **Could lead to biased press based upon financial considerations.** Does anybody find it odd that magazines that cater to Microsoft or Apple products rarely give bad reviews to their respective products? Gee, could it be because 85% of the ads come from those companies or ones closely related? As the adage states, don't bite the hand that feeds you. Go back to when Vista was first introduced and the only bad articles you'll find were in publications not floated by Microsoft. Most current Linux articles are honest because there is no vested financial interest in the end result.



• **Could destroy individulaity.** Will Slackware, a Linux variation created for friends of the developer be able to survive outside of its core element? Will Yellow Dog now advertise itself as the choice of Mac users and thereby dilute its hard core fan base?

• **Might destroy the free advantage currently enjoyed by Linux users.** Would OpenOffice survive if Lotus and Corel get in on the act? Would free OS variations exist as we currently know them? Probably, but they currently survive only because the big boys don't see them as a threat. What if OpenOffice became a mainstream hit and no cost? You can bet your last dollar that MS and Apple will retaliate with a vicious counter-attack.

• **Could lead to garbage distributions created as a stopgap to make money.** Anybody remember Me? Created until something better came along, Me nearly ruined Microsoft's reputation. In literary circles this is known as created a potboiler, that is, something to make a little bit of money before the real prize is issued. When it comes to money and the possibility of making more, common sense

often goes out the door for the sake of a quick buck.

• **Might breed contempt through familiarity.** I can remember when Windows was introduced in the 80s and it was seen as the answer to all of the computer ills to date. Now they get more grief than praise because of ill prepared products pushed out to the public. Could Linux end up in the same category one day? Remember that Saturn started out as GM's savior and ended up the pariah.

• **Is destined to make #1 paranoid.** And that may not be pretty. If MS and Apple feel threatened, the legal fangs will show. It wasn't so long ago that Microsoft was suing Linux developers over patent infringement, although that furor died down when Microsoft couldn't quite tell everybody what patent was being violated. Apple has spent millions suing Psystar over improper distribution of their (Apple's) OS on non-Apple computers. These guys didn't make it to the top by playing nice.

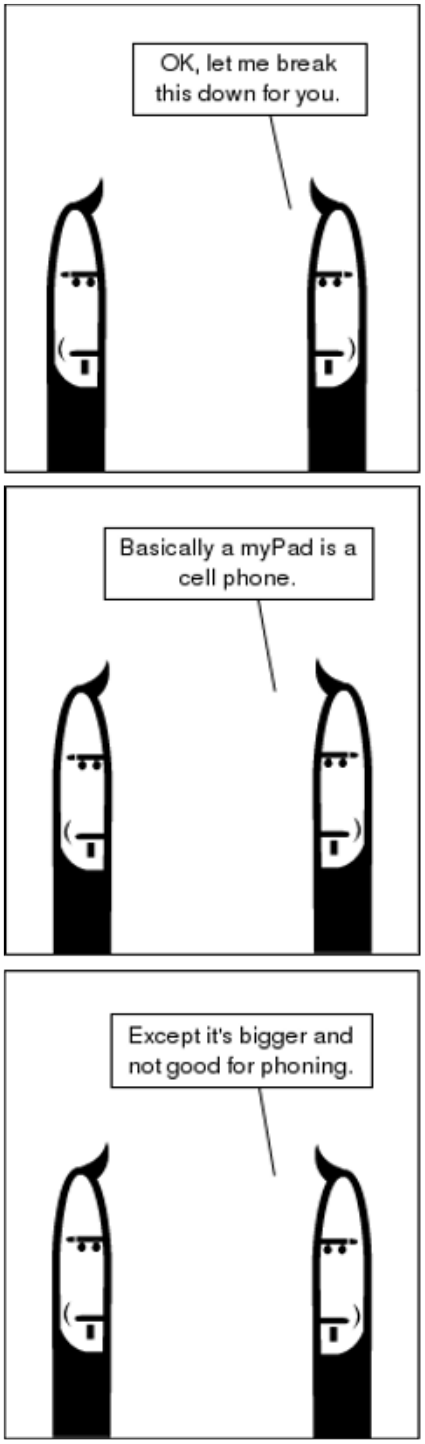
• **Could polarize issues based upon money more than sense.** Most in the Linux community see common issues that should be addressed such as compatibility, boot speed, program updating, etc. Once in the mainstream the issues become clouded by monetary concerns. Why speed up the boot process if the current system is equivalent to whatever else is offering? Why upgrade when consumers are still buying the old version? The relevant issues get pushed aside for the sake of a profit and that's when quality goes downhill quickly. Too many companies have failed because they merely whitewashed an existing product and hoped the consumer didn't notice nothing was new.

• **Could make Linux developers forget the public is a fickle mistress.** If you become popular and intend to remain that way, be prepared to update frequently. Look what happened when Microsoft discontinued XP for Vista. Some users found the new OS cranky and unreliable only to discover XP was being discontinued by Microsoft (who then had to backtrack and continue support for its 'discontinued' product). As it currently stands, many Linux developers update on a consistent basis with Ubuntu getting one every six months like clockwork. Make it a pay-for-use system and see if customers start demanding instead of anticipating updates.

Is there an easy answer? No, and I seriously doubt we'll ever have one. Crossing that perilous passage to a mainstream OS is a risky proposition that is fraught with pitfalls. Tread lightly and carefully.

Is being secondary to Apple really that bad? That corporation has made a decent living off being #2 and from what I've seen of recent stock prices, they are actually doing better than Microsoft. Apple is an exclusive club and they pick no bones about it.

Question is, should Linux stay that way? **You decide.**



Motorola Milestone / Droid

2G GSM 850 / 900 / 1800 / 1900
3G HSDPA 900 / 2100
115.8 x 60 x 13.7 mm
Weight 165 g
TFT touchscreen, 16M colors
480 x 854 pixels, 3.7 inches
Multi-touch input method
Accelerometer sensor
Proximity sensor for auto turn-off
Full QWERTY keyboard
Vibration; MP3, WAV ringtones
3.5 mm audio jack
133 MB storage, 256 MB RAM
microSD, 8GB included
WLAN Wi-Fi 802.11 b/g
Bluetooth
microUSB v2.0
5 MP, autofocus, dual-LED flash
Video (720x480 pixels)@24fps
Android OS, v2 (Eclair)
ARM Cortex A8 600 MHz processor
Digital compass
MP3/eAAC+/WAV/WMA9 player
MP4/H.263/H.264/WMV9 player
Adobe Flash Player v10.1

Cost: ~£380

I'm sure you all remember the original 'Google phone', the G1. Now, Motorola has created a sibling to the G1, the Motorola Milestone (or 'Droid' if you're stateside) and not only is it more slimline than the G1, it also comes with Android 2.0! Android is, of course, based on our beloved Linux, hence why I've included this review in a Linux magazine.

The first thing I noticed about the Milestone is the phone and slider feel rock solid. The Milestone slider needs a good shove all the way to click in place, and I like that.

On putting my 3 SIM in (the Milestone is unlocked) and powering up the Milestone I was greeted with the 'android' loading screen then the familiar Android desktop, now it's time to log in with my Gmail info. I was interested to see if everything will still sync OK. Technically it should, of course, but I'm old school and used to having to manually enter all your contacts on getting a new phone/SIM. Once signed in, the phone connected to Google and all my Gmail contacts are there, same with my Google calendar stuff.

One thing that took a bit of getting used to was having no buttons on the front of the phone. There are four icons below the screen. They are actually touch sensitive buttons which come in to play when the screen is active.

Having tinkered with several home screen replacements on the G1 (GDE being one of my favourites) I've grown to love SlideScreen (above) so, after signing in, I headed to the Market to install SlideScreen, but would the Market still allow it? After all, I paid for SlideScreen from my G1. But, of course, everything is saved at Google's end, so it does indeed mark it as paid and installs in a jiffy. The Market also remembers several other apps that I recently (un)installed under the Downloads tab, so that's quick links to apps that I need.

One thing I did notice more on the Milestone is that up top where it normally shows the 3G logo, I was seeing an 'H'. Turns out this is some super-fast 3G connection which I never got on the G1. Another bonus!

My next port of call was to install K9 for my non-Gmail needs, but I noticed an icon for Gmail and Email in the system settings. Prodding email lets me set up a non-Gmail account without the need for another app! There are also icons for

Corporate Calendars and what-not, allowing you to sync with Microsoft Exchange. Which is no doubt very welcome for business users.

I installed several of my other apps (Camera ZOOM FX, Droid Analytics and Doodledroid to name but a few, but see this month's Top 5 for my top Android apps) and they certainly seem faster on the Milestone. I installed Camera ZOOM FX as I could never seem to get a decent picture with the standard camera app, Camera ZOOM FX uses the sensors within the phone to detect stability before taking a photo. And the resulting photos are good, it's now a 5mp camera.

I also kept wondering why the screen was fading in and out. Turns out the Milestone has a sensor at the top left to detect the light level where you are, and adjusts the brightness of the screen accordingly, so when I pick up my phone in the dead of night and turn it on, it will no longer blind me. Another nice touch!

Since my iPod is on its last legs, I installed MixZing (pictured) and copied across several MP3 files. MixZing did a great job of playing the files and sending info to LastFM. Its minimal display shows just the album cover with play/pause, love/hate and so on but also uses your played songs to offer suggestions for new stuff! The good thing with the Milestone, compared to the G1, is the standard ear/headphone jack at the top of the unit, so this now becomes a phone/camera/mp3 player.

Next, Twidroid. I install it because it works well with SlideScreen and Twidroid will also let me try out the keyboard. Like I said, the keyboard is revealed with a resounding click. Typing is easy enough although it would probably have been better if each key were raised slightly. On the Milestone all the keys are little flat squares, unlike the G1 where each key has a gap between it and a bevel to each button. Although why they bothered with the black rectangle to the right of the keys is anyone's guess, it's basically a D-pad like you'd get on a SNES controller. Only downside with the keyboard is that you need to press ALT to get numbers, I reckon Motorola should have let the screen slide up another 10mm and given the number keys their own row at the top of the keyboard.

Google has also incorporated a lot of voice activated stuff in Android 2.0. After several tries, it got my voice commands right. So it might be an idea to incorporate some sort of 'voice training' thing in Android so that the OS can get used to accents.

The Motorola Milestone doesn't come cheap. Mine cost me about £340 on eBay, but it's definitely a great phone, I can see now why everyone was going droid-crazy over the pond. It has more internal power and storage than the G1 so it doesn't feel like a new phone, it feels like a step up, the next-gen if you like. Same with Android 2.0, it's no major update, but it's enough that you notice the extra features (voice commands, non-Gmail etc.) without making you relearn the UI. It's expensive, but I love it, it's definitely my new all-in-one phone/camera/mp3 player gadget.

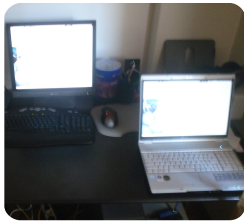
Rating: 9/10

MOTU INTERVIEW

Taken from behindmotu.wordpress.com

Pedro Fragoso

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: 23
Location: Lisbon/Portugal
IRC Nick: ember

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

My first distro was Mandrake, I don't remember which one, but I remember that i started using Linux when Red Hat 6.0 was released (that was my second distro). I've used Red Hat up to 8.0 and then switched to Slackware -> ArchLinux -> Gentoo -> Debian -> Ubuntu. (sort of)

How long have you been using Ubuntu?

I've switched to Ubuntu when Gutsy was still in development, I kinda installed Feisty and upgraded to Gutsy.

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

When Gutsy was released I decided to take a shot and get into development for Hardy, I spent my Gutsy journey trying to understand some bits of Ubuntu development. I started to contribute some bits and giving some love to packages of the Desktop Team.

What helped you learn packaging and how Ubuntu teams work?

Mostly by reading the wiki.ubuntu.com and asking on IRC. And with help of some people (like Daniel Holbach and all the Ubuntu Desktop Team)

What's your favorite part of working with the MOTU?

To be part of making something great like Ubuntu, the community and all the people around Ubuntu development.

Any advice for people wanting to help out MOTU?

Read, start with simple things like triage bugs, and then try something like patches, packing, merges, new upstreams, if you need help spend some time on irc (#ubuntu-motu) and ask.

Are you involved with any local Linux/Ubuntu groups?

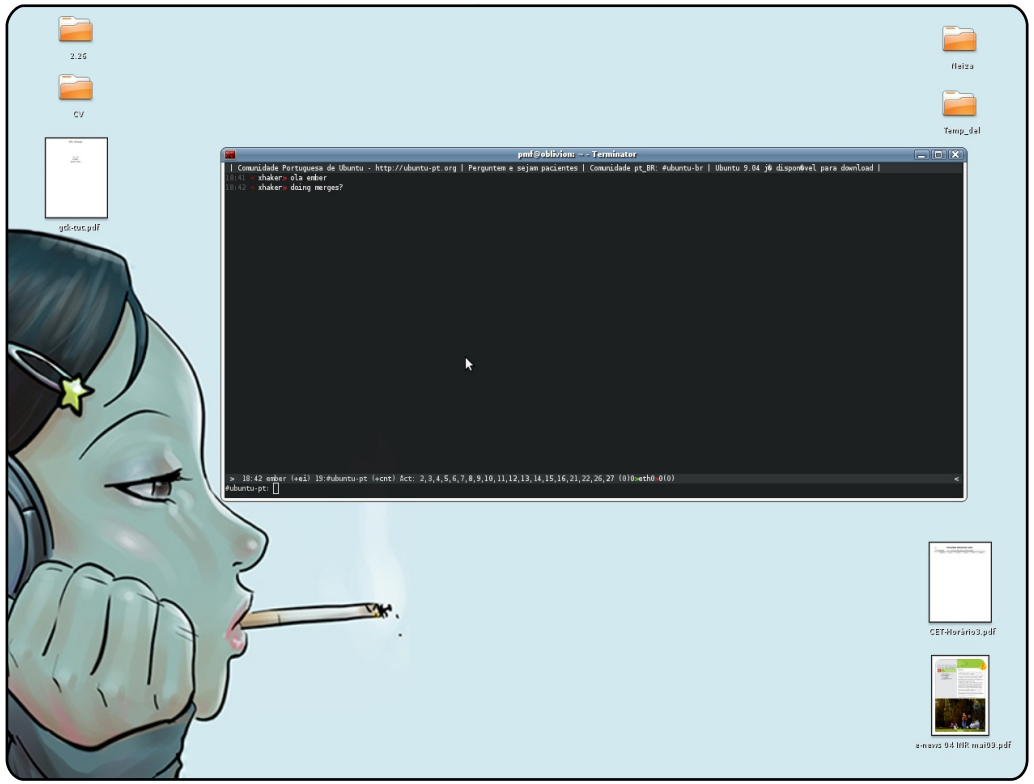
Yes, I'm involved in the Portuguese LoCo.

What have you focussed on in Karmic?

Cleaning some bugs and polishing some gnome updates, and trying to work more with upstream.

What do you do in your other spare time?

Spare time? My spare time is for Ubuntu, the other 'time' is for working/studying and sleeping.



LETTERS

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.

Unison Correction

I've just read your review of Unison in FCM#33, and I have a correction. You write: "*The main issue is that Unison sometimes hangs up with large files, especially over the internet; the standard rsync, in comparison, handles large files just fine.*"

This is true for default non-configured Unison, and for version 2.27.57 that is in the Karmic repository. However, newer versions provide options for using rsync for large files (http://www.cis.upenn.edu/~bcpierce/unison/download/releases/stable/unison-manual.html#speeding). This way, Unison gets all advantages of rsync, and the above mentioned critique is addressed.

This does require that the user fetches Unison 2.32.52 and builds it themselves.

Kristian Kjærgaard

BOS (FCM#29)

BOS is a real time strategy game. The game acted as a catalyst for the student's communication because of his interest level. The game itself is just a game, but the student became so enthusiastic in regard to the game that he desired communication about it. There is nothing special about the game, nor is it actually intended as an educational tool; it just happened to pique his interests. Previously, the student manifested typical behaviors for a child with the autism diagnosis. I attempted to engage him in a myriad of ways and through a plethora of media. After spending time with the student I noticed he would choose computer games as a “choice” activity and from there I exposed him to an array of games that I hoped would interest him. The BOS strategy game was the one

LETTER OF THE MONTH

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

Why does Ubuntu force us to jump through hoops just to connect to the internet via dial-up? Any miserable Windows computer can do that and the process with them requires hardly any thought!

It's not my computer which is the problem, it's the US Robotics external modem (which does work with Linux, specifically the modified Xandros found on my Asus EEE PC 1000 netbook) which I can't get working with Ubuntu 9.10 and earlier on my Acer.

The US Robotics does "light up" (in other words, the computer recognizes that it is there) but I can't connect to my EarthLink dial-up account. I have tried to understand the instructions given in various Ubuntu Forums posts but it just won't work!

Until I can reliably get a dial-up connection using Ubuntu, I am forced to continue using this Asus netbook with its otherwise inferior operating system when I travel.

Surely someone out there knows exactly how to connect via dial-up (there must be someone!) who would be willing to write an article for the magazine. A step-by-step tutorial on to how to simply connect to a dial-up network via Ubuntu would, I believe, be of great service to your many readers.

Lawrence H. Bulk

that earned his attention. He became enthusiastic about it and would initiate conversations on the game, which allowed me to work with him on social skills and develop appropriate communication skills. Once he had become less introverted he was able to create his own story lines, orally, for the characters and of course this led into writing and art. I have worked with this child for three years and it was the games for Linux that I found motivated this child's social development. We also had a great success with GCompris, Childsplay and chess. I found that when his "left brain" was engaged, his "right brain" benefited. I hope this helps clarify how my student made progress.

Anthony Parr

GIMP Clarification

The following quotation is from your editorial: "Speaking of GIMP, we're still getting letters from Ubuntu users who are wondering what, if anything, will replace GIMP in future editions of Ubuntu since surely it needs an image editor of some sort, and I have to agree." And in the letters section Chris Burmajster also wonders what they will replace it with.

I haven't seen any suggestion that Gimp will be completely removed from Ubuntu. The only statement I have ever seen from them is that it will not be included on the installation CD, in order to make space for something with a more general appeal.

I believe it will always remain in the repository for download by those who want to install it. Maybe you could confirm this with Canonical?

Thank you for the magazine. I always enjoy reading it and trying out the tutorials.

Robert Holm

Ed: *GIMP will, of course, remain in the repositories for those who wish to install it but (as was mentioned in episode 2 of the Full Circle Podcast) why have no audio or image editors by default, but (from Lucid onwards) include a video editor? Surely more people edit images than video.*

Restricted Time

One reader asked for the ability to restrict access time to his computer for children. Your response was to use iptables but that, as you said, is an extremely steep learning curve. A better alternative is to use the package timekpr. You can find the package here https://launchpad.net/timekpr. The PPA information for Ubuntu is here: https://launchpad.net/~timekpr-maintainers/+archive/ppa.

This application works very well and is easy to setup for individual user logons.

John Savage

Amber Graner Interview

Penelope Stowe: Please tell us a little about yourself.

Amber Graner: First, thanks for asking me to do the interview. I have to say I am usually the one on the other side asking the questions so this is a fun twist. I am an Ubuntu user advocate, active Ubuntu community member, Ubuntu User magazine blogger and contributor, and event planner as well as a wife and mom. I am quirky, energetic, loquacious, driven, and funny.

PS: You strongly self identify as a "non-technical end user" or NTEU, do you find this makes you unique in the Ubuntu Community? Do you think there's a potential that you will move from "non-technical" to technical?

AG: Nope, as I am not the only person who self identifies as an NTEU, however, I think it is only a perspective though. For example, my husband is someone who has worked with Open Source/Linux companies since the early 90s, so compared to him I am not technical, however when I visit other family or friends who may not know what Ubuntu is, then many times I am the technical person. Would I like to become more technical? Yes, but not because I want to become a "developer," but because I would like to know how and why Ubuntu and the applications I use daily work, so I know where to find information and become a better user advocate when people ask me questions about Ubuntu and what or how they can get involved.

PS: You've recently become the leader of the Ubuntu Women Project, what would you like to see happen with the project under your direction?

AG: As the UW Project leader, it is important to me that I stay focused on insuring the direction and goals of the team are kept on track and that we as a group have continual movement. I feel strongly about making sure we have regular recurring meetings, helping to identify new goals for each release cycle to accomplish the long-term roadmap goals. I am also focusing on the leadership election process that will take place after UDS-M. I want to make sure the terms, responsibilities, and procedures for these yearly elections are in place. These team elections will help the UW Project identify where we can improve, and help other team members recognize their potential as leaders. Through these initiatives the visibility of the UW Project will increase. More importantly, it should also increase the visibility of the contributions of women within and outside of the Ubuntu Community in order to provide examples, role models and mentors to help more women become involved in the Ubuntu Project. This visibility of women within the community will help form a cohesive team and network of women who become stronger users, contributors, developers, advocates and voices within the Ubuntu Community. Seems like a lot, but in just the short period of time since UDS-L, we are meeting about every two weeks, discussions for -M goals have started, and at the end of USD-M the election process should kick off –I think that speaks more for the strength UW Project team than it does about me, as we have an awesome team in place!

PS: In your blog you discuss how the phrase "Linux for human beings" is what brought you into trying Ubuntu, what is it that you think makes Ubuntu do this so well?

AG: I have to admit when I first heard "Linux for human beings" I laughed! I heard people telling me for 15+ years, "oh this is easy," ha! Not for the mere mortal end user like me it wasn't. I truly believe the "by the techie for the techie" days are gone and Ubuntu is changing that.

The fact that average Ubuntu end users don't have to use the command line, know all the technical jargon, or even how it all works is a great benefit. It's not perfect but neither is any other OS. However, Ubuntu is more than a distribution: it is a community as well.

PS: You've done quite a lot in the years since you started using Ubuntu, is there anything you haven't done that you'd like to try?

AG: Hmmm, that's tough, I come across stuff every day that I want to do, but I am aware that I can't take on any more until I have handed off to others some of the things I am currently doing. I want to learn more about the "opportunistic developer," I'd like to learn how to write something so I understand more about it. I want to learn how to write scripts as well. I would like to figure out how to triage bugs and spend some time testing development releases. I want to highlight more community people/loco teams through various interviews. Oh and the list grows, but in the end it is the Ubuntu Community and the average end user that hold my interest and I want to improve how to encourage more people to use and contribute to the Ubuntu Project/Community.

PS: Outside of your Ubuntu work, is there any women-in-open-source or open source work you do?

AG: I help plan Linux Fests, currently working with Atlanta Linux Fest, Southeast Linux Fest and others. I am not a member of any other WIOS groups, not because I am not interested, there just aren't any near where I live, but I do try to speak to members and other leaders of those groups when I am at events so that I can learn from their initiates and experiences.

I contribute to the newest Linux New Media publication Ubuntu User magazine on the You-In-Ubuntu blog (http://www.ubuntu-user.com/Online/Blogs/Amber-Graner-You-in-Ubuntu) and print articles. (http://www.ubuntu-user.com/Magazine/Archive/2010/4).

PS: What's the best thing that's happened to you because of your work with Ubuntu?

AG: First and foremost, I became a Linux user and became an active member of the awesome Ubuntu Community! Aside from that it would have to be working with Ubuntu User and Linux Pro magazine's and reviewing *The Art of Community* by Jono Bacon.

PS: Do you have any other interests or activities you'd like to tell us about?

AG: I have been invited by the *American Dairy Goat Association* to have an Ubuntu booth at the *Goat Festival and Parade* in Spindale, NC. I am hoping the NC LoCo team will want to participate and maybe we can even have a float - who knows. This is going to be a fun new way to introduce Ubuntu to my small area of the world.

UBUNTU GAMES

Written by Edward Hewitt

GAME NEWS

FlightGear 2.0 is out! - Dramatic new 3D clouds and lighting conditions. Also many new and detailed aircraft models.

Commercial games on Linux are very few and far between. When Enemy Territory: Quake Wars (ETQW) was announce for Linux, we were all surprised to see such a big PC game coming to the Linux platform. Developed by Splash Damage with help from id Software, ETQW follows on from the very popular Enemy Territory Wolfenstein, a free closed source game from the same developers, which was a huge hit on the platform. This time round, Splash Damage created the enemy territory game in the Quake universe, but this time at a price for entry.

If you're unfamiliar with the Enemy Territory series, essentially it's a FPS which is played out on large maps. Two teams are battling it out on these maps, but instead of just killing the enemy, there are also objectives which the team will have to complete in order to win the game. These range from destroying or capturing certain points (enemy will have to defend the point), delivering an item to a certain point (enemy will have to make sure they don't deliver it) or open a series of gates to move a vehicle through (enemy will have to stop it). If you have played Wolfenstein ET, it is pretty much the same. If you have not, download it now! It's a free download, and it's one of the best games on Linux.

Quake Wars follows the same game mechanics as the first ET, and it works very well. The objective system is a great way to keep you involved with the game. It can be boring just shooting at the enemy or trying to find where they are. With the objectives, you always know where to go and what needs doing next. You gain a sense of progression. Tied in with the objectives system is the class system. On entering a match, you must select a class to play as. There is a choice of 5, same classes on GDF (Humans) and Strogg sides. Solider is the assault class, there you aim to defend and capture points. Field Ops provides more heavy fire power support. Engineer plays a support role in constructing and repairing vehicles, machinery and bridges. Covert Ops plays a more stealth role with the aid of a sniper rifle. Finally, Medics heal and revive other players. All these classes are very important to win a match, since they all play a role in completing objectives. What is surprising with this game's class system is that all the classes have very powerful weapons, usually you will find the medic and engineer class

to have poor weapons. Tying all these mechanics together is the rank and experience system. With every kill, objective completed and match won, you gain experience. These are all tied into a central stats page, where you can see tons of stats about the classes you play and how successful you have been on the battlefield. I have to mention that one huge improvement over Wolfenstein is the increased map size, which is huge now. This now allows gamers to drive vehicles, such as boats, planes, tanks, jeeps, helicopters and quadbikes.

ETQW is entirely designed around multiplayer. There are no single player missions, except for playing against bots. The online community is well established with hundreds of active servers. It also makes an excellent LAN game as well, with excellent support for LANs. There are plenty of maps, with varied locations and objectives. The game looks stunning, the textures are of high quality. The lighting and particle effects are very impressive. All this comes at a cost to your computer's hardware.

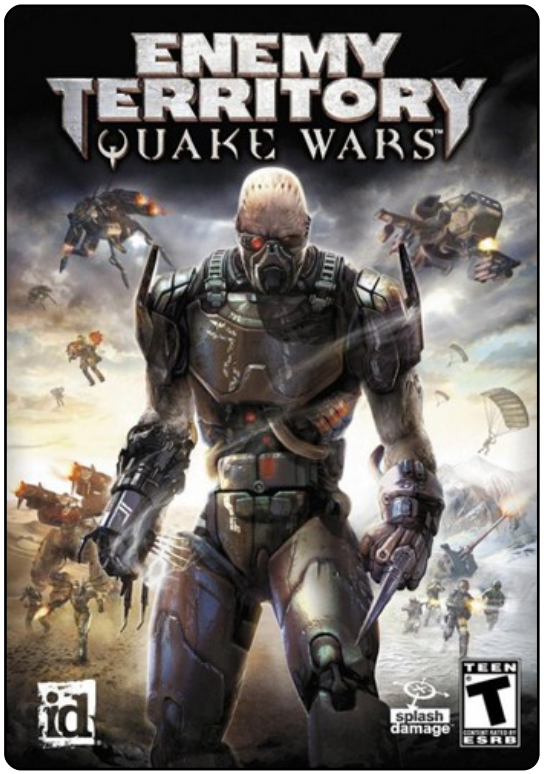
With this review, I could easily retitle it a review of Enemy Territory Wolfenstein. Both these games are essentially the same. What makes Quake Wars stand out is the maps, sound, graphics and objectives. I see Wolfenstein as the demo to the actual game, which is Quake Wars. I urge everyone to pay the price of admission and play Quake Wars. It has been difficult for me to find a fault with the game, the only thing I might complain about is the awkward installation, which requires you to buy the Windows Disc of the game and download the Linux installer. Quake Wars is just too good to complain about, and it's not difficult for me to say it is the best game on Linux, hands down!

Score: 10/10

Good:
- Objectives and Class system
- Excellent Graphics and Sound
- Great maps and online community

Bad:
- Awkward Installation

System requirements:
2.0 GHz processor
512 MB RAM
128 MB video card
Internet connection



How to Install:
You will need the DVD copy of the game and download the Linux client from http://zerowing.idsoftware.com/linux/etqw/

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 Pidgin)

Q&A

Written by Tommy Alsemgeest

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.**

Q A few weeks ago, I updated using the Update Manager. I reboot as requested, but I couldn't get back into Ubuntu. I asked for help around the forums, but had internet access problems. Finally I decided to reinstall using the Live CD as suggested (with some kind people's advice). But I lost a lot of time & effort.

A Each update is stringently tested by either the Ubuntu development team or one of the MOTUs before it is allowed into the main repository, with the update generally going into the "proposed" updates section for people to test. Unfortunately though, it is impossible to try it with everyone's configuration, so the best thing to do in a case like this is to submit a bug report so that (all going well) it will not happen again.

Q I've tried a few low cost bluetooth dongles and discovered that they work with Windows but not Ubuntu. It seems that I was using dongles which used firmware by CSR, which is apparently unsupported, or buggy, due to a bug that is published but has "unassigned" state.

Most of low cost dongles seem to be using CSR firmware which seems to be a favorite with low-cost dongle makers, so it is becoming that much more difficult to locate a working dongle in India.

Is this as big a deal to get a working dongle as it is appearing to me?

A My cheap little bluetooth dongle has worked out of the box with Ubuntu since I bought it, back with Ubuntu 7.10. The unfortunate thing is that if it doesn't work straight away, your chances aren't good of getting it working. If you browse wiki.ubuntu.com and search for bluetooth usb support, it will take you to a list of USB adapters, with comments about whether they work in Ubuntu.

Q For the last 2 months I have been getting "your session lasted for less than 10 seconds." So I haven't been able to log into my Ubuntu.

A If you can log into Recovery mode (should be an option when booting,) you can go into the root shell, and create a new account to log into. You can do this with these two commands:

```
useradd <username>
```

```
passwd <username>
```

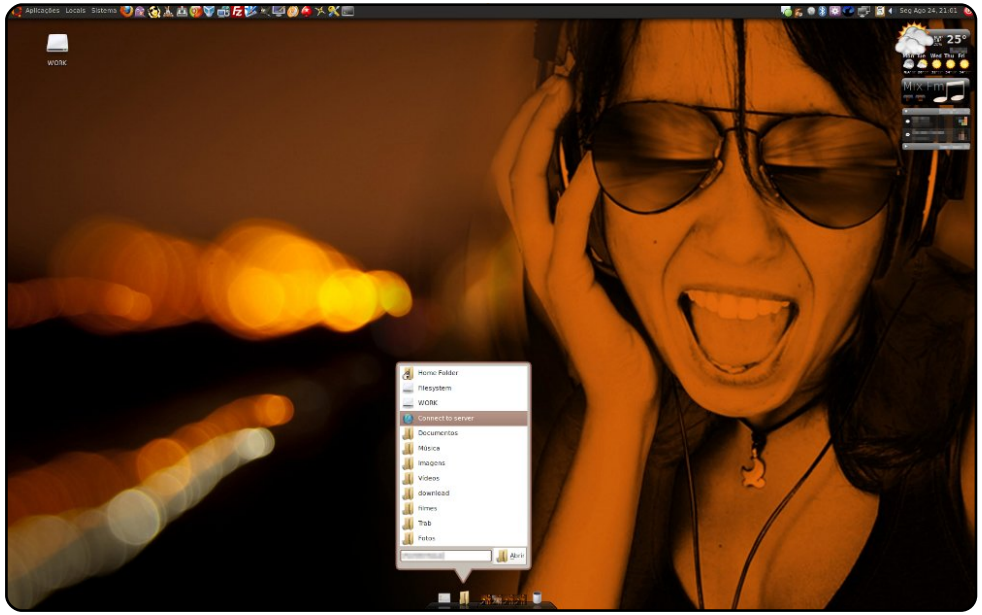
Replacing <username> with your new username. Reboot and try to log into your new account. Unfortunately, if this doesn't work, your best bet is probably to back up your data and reinstall Ubuntu.

Q I run Kubuntu 9.04 with KDE 4.3.1. Every time I try to open files and browse my local disk (from File > Open) in an application, the application will crash. This happens in Firefox and Comix, and recently I found that it happens when I try to export in Audacity. When I try to run Comix from the command line, it will crash and close, and I got a message 'Segmentation fault'.

A The most likely cause of a segmentation fault (segfault) is faulty hardware. Boot from the Kubuntu Live CD. If it doesn't segfault, the problem isn't a hardware problem and could be fixed with a reinstall. If, however, the same thing happens, you must replace the faulty piece of hardware.

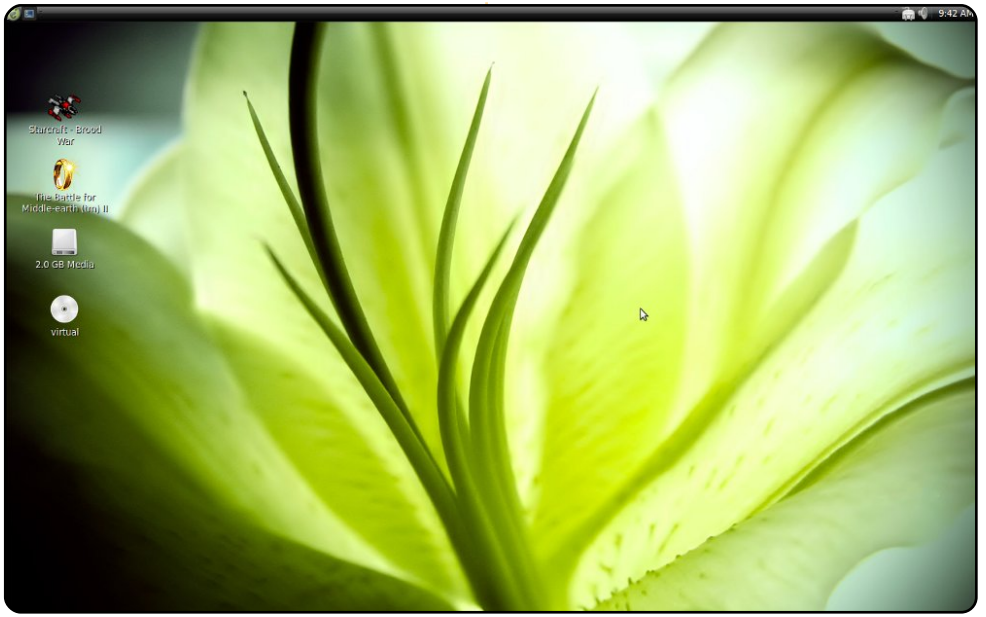
MY DESKTOP

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.



CPU: Intel(R) Core(TM)2 E7400 @ 2.80GHz
Ram: 4Gb
System: Ubuntu 8.04 (I'm having hardware trouble in 9.04)
Theme: Dust
Icons: nuoveXT-1.7
Screenlets: ClearWeather; Radio (with modified skin by myself); Pidgin (with modified skin by myself)
Avant Window Navigator Applets: Show Desktop; File Browser Laucher; Shiny Switcher; Stacks Trasher
Wallpaper: "01762_lights_1920x1200" (http://wallpapersbrasil.net63.net) mixed with "Dj by Frenzyy" (http://frenzyy.deviantart.com/art/Dj-49068185) used with Xara Xtreme

dj mau'z



I am a 19 year old Linux enthusiast, and I live in the United States. I am known as *NightwishFan* on Ubuntu Forums. The first Linux system I tried was Ubuntu Gutsy, which went beyond my expectations of what I then thought a free software OS could accomplish.

My current desktop is an aging Compaq with about 1gb of ram, and 64-bit Ubuntu Jaunty, which entirely replaced Vista. After two years of using KDE, I have recently gone over to Gnome. I customized my desktop to be very lightweight, with no unneeded startup services. I also have the panel and Metacity configured to give my applications more screen space, much like the netbook remix. My themes and wallpaper all come from the bisigi repository: http://www.bisigi-project.org/?lang=en

Shimizu Kazuma

My name is Putu Gema Bujangga. I'm a geologist, from Bali, Indonesia and now work as a coal geologist. I dual boot my laptop with Ubuntu Jaunty and Windows XP. Some of the geology-engineering programs (such as Surfer, Logplot and Mincom Minescape) make me still use XP. Some experiments to install those with WINE result in failures. Nevermind that. I've been using Ubuntu since Hardy and will love Ubuntu 'till death. :p

I installed Ubuntu on my Compaq C733 with Intel 540 1,86 Ghz. To provide a clean and elegant desktop environment, i'm using Sysmonitor from Screenlets (you should try it with Conky too, the result is nice). The rest is common GNOME Panel 2.26 with font 60 pixels. Here you go.

Putu Gema Bujangga

My name is Ahmed Abd Alatif, I am 20 years old and I have been using Ubuntu for three years now and use Ubuntu 8.10 with Compiz Fusion and AWN with Google Gadgets on my asrock 775i56, with Intel dual core and 512 MB of ram. This is the Ubuntu sunrise theme.

Ahmed Abd Alatif

Thinking Space

http://www.thinkingspace.net/

Thinking Space is a mind mapping application for the Android OS. Portable mind mapping has never been so much fun! Thinking Space is available for free in the Android Market, just search for 'Thinking Space'. Not only is the mind map stored locally, but you can also have it export images to email, or store the map in a cloud for sharing with other users.

WordUp!

http://www.anthrological.com/

It's fun to unlock doors, discover ancient secrets, and dispel urban myths. Hey, it's also fun to find words hidden in a grid! This game is quick, fun, and you'll end up being smarter than your friends. On the grid, trace your finger over letter cubes to spell out words. Choose between 4x4, or 5x5 grids. Head over to the Android Market today to get your copy.

PicSay Pro

http://www.picsaypro.com/android/

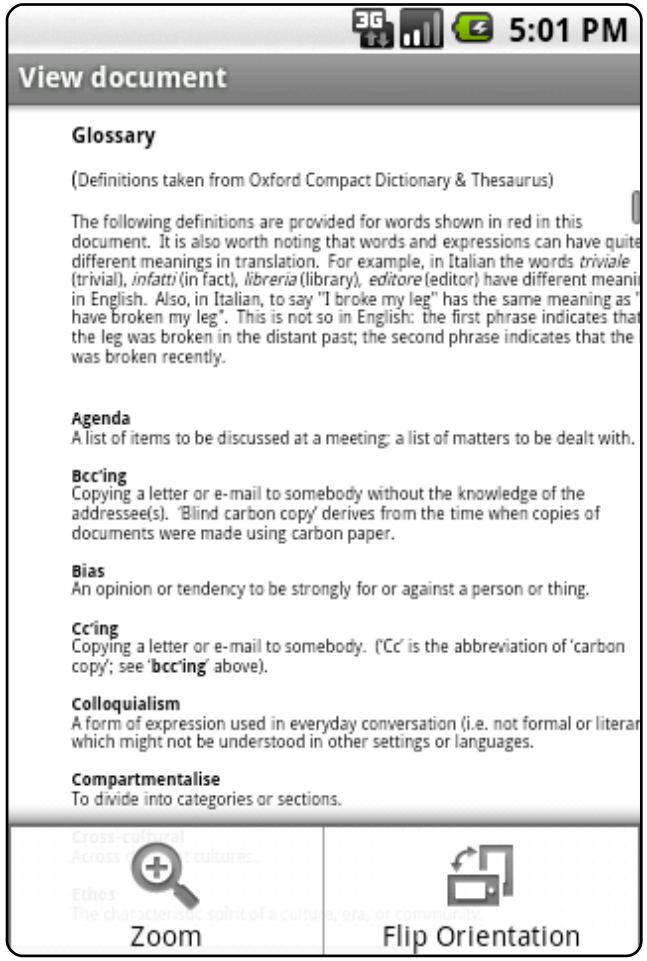
PicSay Pro is a powerfull all-in-one photo editor for Android. It provides a host of features in a fast and easy-to-use interface. PicSay has both free, and paid options. I usually use it to edit the colors of photos taken with my phone's camera, but it's also nice for adding reminder text, even in the form of speech balloons.

GDocs

http://www.gdocs.mobi/

GDocs is an editor/viewer for your Google Documents. It is an Android application that allows you create, edit, view, import, export and send documents as well as sync documents with your Google Docs account.

Very handy, this little app lets me edit FCM articles on the move!

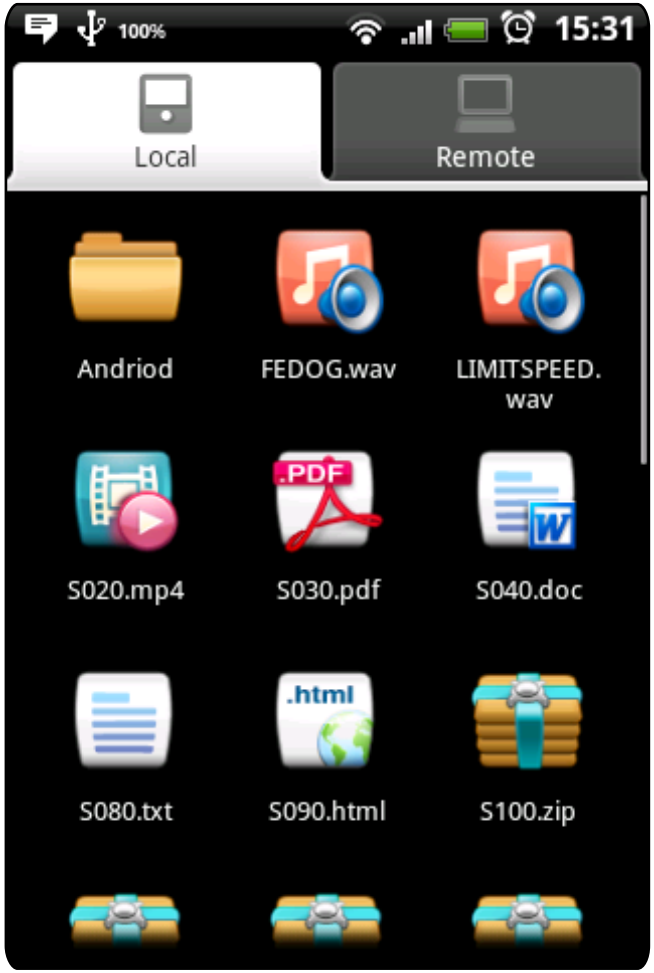


EStrongs File Explorer

http://www.estrongs.com/

EStrongs File Explorer is a featured File/Application Manager which can explore the phones and PCs in a LAN via the Samba protocol.

I've tried several file managers in Android and this one is, to me, the best of the bunch. Since it also allows remote connections via LAN, and wireless, it's very handy.



The Ubuntu UK podcast is presented by members of the United Kingdom's Ubuntu Linux community.

We aim is to provide current, topical information about, and for, Ubuntu Linux users the world over. We cover all aspects of Ubuntu Linux and Free Software, and appeal to everyone from the newest user to the oldest coder, from the command line to the latest GUI.

Because the show is produced by the Ubuntu UK community, the podcast is covered by the Ubuntu Code of Conduct and is therefore suitable for all ages.

http://podcast.ubuntu-uk.org/

Download

Available in MP3/OGG format in Miro, iTunes or listen to it directly on the site.

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

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

... or you can visit our **forum** via: 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
Podcast - Robin Catling
podcast@fullcirclemagazine.org

Editing & Proofreading
Mike Kennedy
Robert Orsino
Gord Campbell

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

Deadline for Issue #36:
Sunday 11th April 2010.

Release date for issue #36:
Friday 30th April 2010.