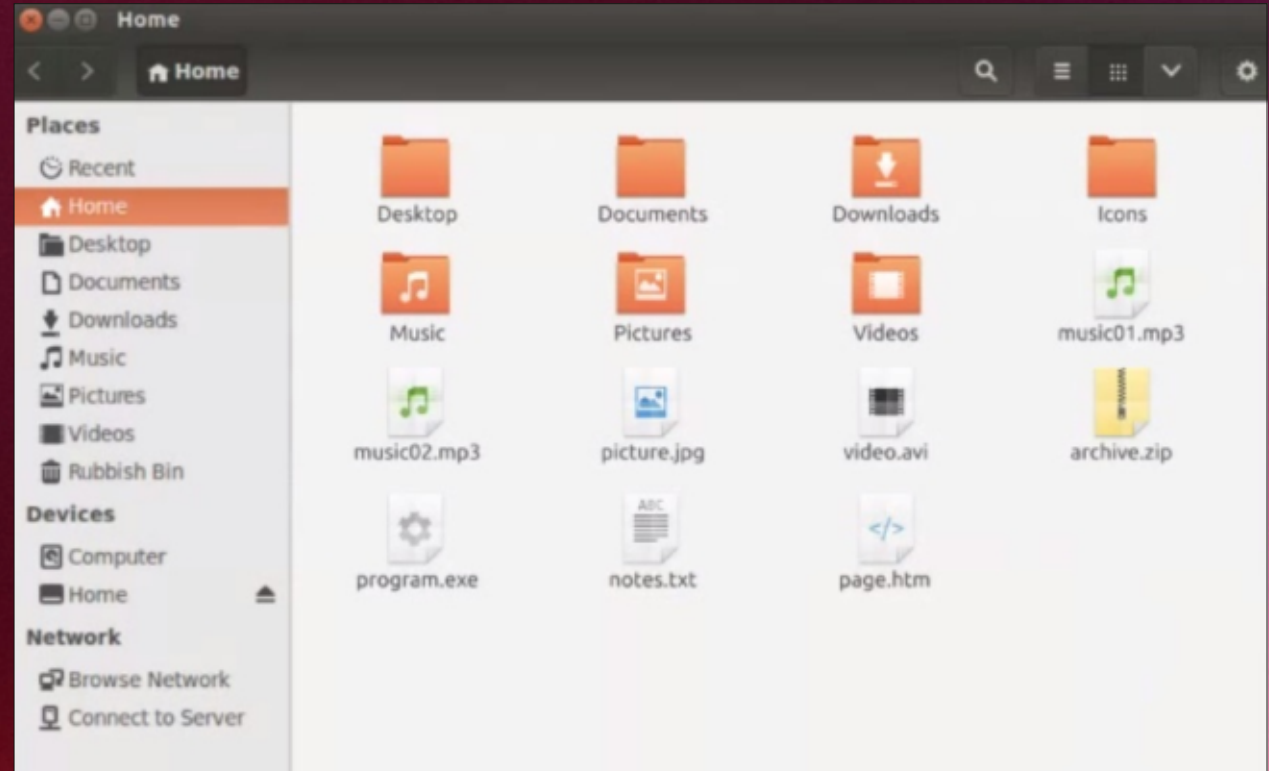




Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

ISSUE #85 - May 2014



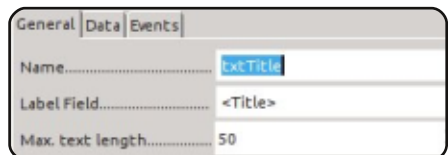
UBUNTU 14.04 UNDER REVIEW



HowTo



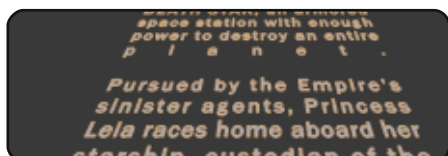
Python p.11



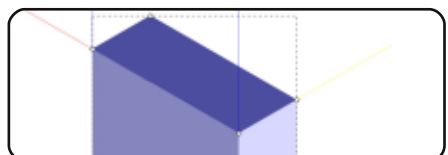
LibreOffice p.14



GRUB2 & Multibooting p.17



Blender p.19



Inkscape p.21

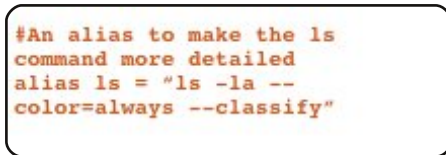


Graphics



Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY



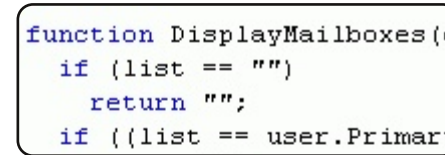
Command & Conquer p.09



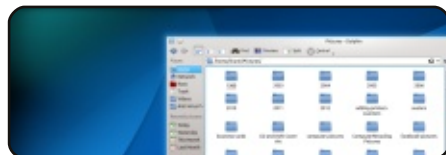
Arduino p.25



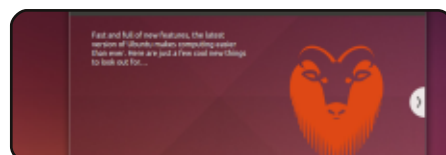
Linux News p.04



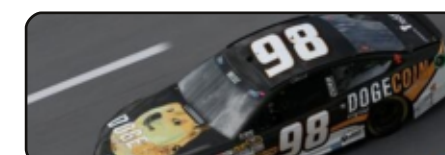
Ask The New Guy p.29



Linux Labs p.33



Review: Ubuntu 14.04 p.37



CryptoCurrency p.39



Letters p.42



Tuxidermy p.43



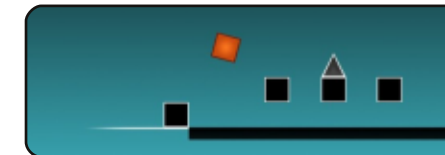
Q&A p.44



Security p.46



Open Source Design p.47



Ubuntu Games p.49



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

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



WELCOME TO ANOTHER ISSUE OF FULL CIRCLE.

It's a full house this month. Greg's Python article is here beside Elmer's LibreOffice. Completing the set is the first of several parts explaining the GRUB boot menu and multibooting. Inkscape and Blender are here too, and the first of my Arduino series of articles. In this first one I'm just tinkering with some sensors, but, from next month, I'll be going through the trials and tribulations of my first project.

Our review this month is Ubuntu 14.04 – which was released roughly a month ago. Robin has cast his critical eye over it, and will give you his verdict. Me? I'm still using Mint 16, but Charles seems to have fallen head over heels for my old friend Kubuntu. I'll probably return to it one day; I always have in the past.

Only two My Desktop screens this month as, I'm afraid, that's all I have. If you'd like to show off your desktop to the readership, then feel free to email me a screenshot and some text about how you got your beloved desktop.

I was sent a security question for Michael (which he answers this month) which was something I hadn't considered before. In short: can free virus scanners be trusted? If you have a security based question for Michael, please send it in and I'll pass it along to him.

And finally, Oscar has a couple of Humble Bundles to give away. So, have a read of his review of The Impossible Game, answer the simple question, and you might win some freebies!

All the best, and keep in touch!

Ronnie

ronnie@fullcirclemagazine.org



This magazine was created using :



Full Circle Podcast

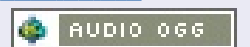
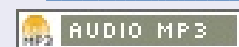
Released monthly, each episode covers all the latest Ubuntu news, opinions, reviews, interviews and listener feedback. The Side-Pod is a new addition, it's an extra (irregular) short-form podcast which is intended to be a branch of the main podcast. It's somewhere to put all the general technology and non-Ubuntu stuff that doesn't fit in the main podcast.

Hosts:

- Les Pounder
- Tony Hughes
- Jon Chamberlain
- Oliver Clark



<http://fullcirclemagazine.org>



Download



ROBOLINUX TURNS YOUR C DRIVE INTO A VIRTUAL WINDOWS MACHINE YOU CAN RUN IN LINUX

Say you want to move from Windows to Linux but there are a few Windows apps that you can't give up, and they don't work well under WINE. The developer of Robolinux offers a Debian-based GNU/Linux operating system designed to let you run Windows XP or Windows 7 in a virtual machine.

The latest version of Robolinux goes a step further: It includes a tool that lets you create a virtual machine by cloning your Windows C: Drive, which means it takes just minutes to create a version of Windows that you can run in virtualization in Linux, and it will already have all of your existing programs and data.

It lets you do it without using a fresh Windows license key. That can come in handy if your only copy of Windows came with your

computer or if you have an OEM license which is allowed to be installed on only a single computer. This tool was developed by Robolinux maker John Martinson, you can also use the software with Ubuntu, Linux Mint, OpenSUSE, Fedora, Debian, or 500 other Linux distributions.

Source:

<http://liliputing.com/2014/05/robo-linux-turns-c-drive-virtual-windows-machine-can-run-linux.html>

Submitted by: **Brad Linder**

FORGET ABOUT RAZOR-QT AND LXDE, IT'S TIME TO EMBRACE THE BEAUTIFUL LXQT

The LXDE and Razor-qt teams are proud to announce LXQt 0.7.0, the first release of LXQt, the Qt Lightweight Desktop Environment. This beta release is considered a stable continuation of the Razor desktop.

The hardworking developers and dozens of contributors and translators, after almost one year since the Razor-qt and the LXDE-Qt project decided to merge, proudly present the beta release of LXQt 0.7.0. The new website for the LXQt project is available at <http://lxqt.org/>.

The existing LXDE users do not have to be worried for the time being, because the GTK version of LXDE will still continue to be maintained as long as there are developers still working on it; several developers have shifted their focus to LXQt. Packages of LXQt are already available for the following distributions: Arch Linux through AUR, Ubuntu with daily builds PPA, Opensuse and Siduction.

Source:

<http://news.softpedia.com/news/Forget-About-Razor-qt-and-LXDE-It-s-Time-to-Embrace-the-Beautiful-LXQt-441162.shtml>

Submitted by: **Silviu Stahie**

DEBIAN 7.5 "WHEEZY" LIVE CD NOW AVAILABLE FOR DOWNLOAD

The Debian project has released the new Live CD images for the latest version of Debian 7.5 "Wheezy," that was made available a week ago.

When a new point release of Debian is made available, the Live CD version of that distro is not accessible to users right away. It usually takes about a week for the Debian Live CD team to put together the new releases. All the Debian flavors have gotten their own Live CD, including LXDE, GNOME, KDE, and the Rescue CD. The default implementation is Xfce.

Source:

<http://news.softpedia.com/news/Debian-7-5-Wheezy-Live-CD-Now-Available-for-Download-441273.shtml>

Submitted by: **Silviu Stahie**

UBUNTU AIO DVD HAS ALL UBUNTU 14.04 LTS FLAVORS ON ONE DISK

Ubuntu AIO DVD (all-in-one), a collection of the most important Ubuntu 14.04 LTS flavors made available on April 17, 2014, is now ready for download.

The Ubuntu AIO DVD was put together by Milan Rajčić and helps users have all the major Ubuntu spins on a single DVD: Ubuntu 14.04 LTS, Kubuntu 14.04 LTS, Ubuntu GNOME 14.04, Xubuntu 14.04 LTS, and Lubuntu 14.04 LTS.

As you can imagine, this is a very large compilation and it holds the official images that you can also download from the Canonical servers. The difference is that users have a single image that holds them all.

Source:
<http://news.softpedia.com/news/Ubuntu-AIO-DVD-Has-All-Ubuntu-14-04-LTS-Flavors-on-One-Disk-441202.shtml>

Submitted by: **Silviu Stahie**

US NAVY IS SWITCHING FROM SOLARIS TO LINUX FOR ITS UAV PROGRAM

The US Navy has contracted a third-party company to help them switch to Linux for their UAV program – not from Windows, but from Solaris.

Solaris has been considered for many years one of the most stable solutions available, whether if it was for military purposes or for civil applications, such as airports and other branches. This is not one of the most publicized operating systems in the world, but Solaris is actually running on a number of critical systems.

“Unmanned aerial vehicle (UAV) control experts at the Raytheon Co. Technical Services segment in Dulles, Va., will switch a major unmanned helicopter control system from Solaris to Linux software, and upgrade the system with universal UAV control qualities under terms of a \$15.8 (€11.3) million contract.”

The new system will have to provide some essential features that seemed to be missing on the current platform, such as intuitive

controls, automated testing procedures, and support software upgrades in the field.

Source:
<http://news.softpedia.com/news/US-Navy-Is-Switching-from-Solaris-to-Linux-for-Its-UAV-Program-441062.shtml>

Submitted by: **Silviu Stahie**

ANTI-VIRUS PIONEER ALAN SOLOMON THINKS ANTI-VIRUS IS DEAD. HE USES LINUX INSTEAD

British anti-virus pioneer Dr Alan Solomon is so convinced that AV software no longer works that he gave up using it a “long time ago” and solved his security worries by moving from Windows to Linux, the iconic figure has said in a blog.

John McAfee rubbing the software that still carries his name was one thing, but Solomon’s more clinical disassembly of an industry he helped create in the 1990s with Dr Solomon’s highly-regarded Anti-Virus Toolkit (bought by Network Associates in 1998 for \$642 million)

is more like a well-aimed punch in the solar plexus.

Describing Symantec’s recent declaration that antivirus is “dead” as a “Gerald Ratner” moment [see endnote], Solomon goes on to deliver the coup de grâce.

“I stopped using an antivirus a long time ago, because I couldn't see how it could work in a world where you would need daily updates, which means that each update is tested for ... how long? Not very long, obviously. Because these days, we're looking at around 100,000 new malware samples PER DAY. Or 200,000, depending on who you talk to.”

Source:
<http://blogs.techworld.com/war-on-error/2014/05/anti-virus-pioneer-alan-solomon-thinks-anti-virus-is-dead-he-uses-linux-instead/index.htm>

Submitted by: **Arnfried Walbrecht**

NEW UNREAL TOURNAMENT IN DEVELOPMENT, AND IT'LL BE ABSOLUTELY FREE

Epic Games announced today that it has started work on an all-new Unreal Tournament game. Interestingly, the Unreal Engine 4-powered game will be entirely free when it's released. Not free-to-play, completely free, Epic Games said. A "small core team" is currently working on the game, which is coming to Windows, Linux, and Mac.

Development on the game began today, May 8, Epic Games said. The studio plans to develop the new Unreal Tournament game "in the open," as a collaboration between Epic Games, Unreal Tournament fans, and Unreal Engine 4 developers. "The game will be true to its roots as a competitive FPS," Epic Games says.

Epic Games cautioned that it will take "many months" until the studio has a build of the game that's playable. "This is real development from scratch," the company says.

Source:
<http://www.gamespot.com/articles/new-unreal-tournament-in-development-and-it-ll-be-absolutely-free/1100-6419511/>

Submitted by: **Arnfried Walbrecht**

RED HAT LOOKS TO BE PLATFORM OF CHOICE IN CLOUD

Red Hat has been instrumental in the evolution of Linux from a geek-friendly, hacker-based open source OS to a full Tier 1 viable enterprise solution. At Red Hat Summit 2014, John Furrier and Stu Miniman of The CUBE talked with Jim Totton, VP & GM of Platform Business Unit at Red Hat about Red Hat as a platform and how it is becoming the platform of choice for on-premise, virtualization and ultimately the cloud.

Source:
<http://siliconangle.com/blog/2014/05/07/red-hat-looks-to-be-platform-of-choice-in-cloud/>
 Submitted by: **Arnfried Walbrecht**

NOW, UPDATE LINUX SERVERS WITH NO DOWNTIME

With KernelCare, now

available from CloudLinux, scheduled outages for security patches on Linux servers are now a thing of the past, giving organizations real-time updates.

KernelCare automatically applies Linux server security updates without having to re-boot, freeing technical personnel from the laborious process that takes several minutes for every server, several times a year.

"This is the equivalent of changing the engine on an airplane while it's flying," said Dan Olds, principal analyst, Gabriel Consulting Group. I think this will be viewed as a no-brainer purchase when you consider the cost of less than \$50 annually per server for having the protection of kernel security updates without downtime.

Source:
<http://internet.itbusinessnet.com/article/Now-Update-Linux-Servers-with-No-Downtime-3236273>
 Submitted by: **Arnfried Walbrecht**

IN THE CLOUD, LINUX WILL WIN THE OS WAR

A recent article in Fortune magazine entitled "The Dawn of the Chrome Age" highlights the success of the Linux-based OS in the low-cost laptop market. According to the article, "Over the holidays in 2013, two Chromebook models were the No. 1 and No.3 best selling laptops on Amazon.com, and they're being adopted in schools and business around the world." Simply put, Chrome OS represents Web apps on top of Linux, and given that the Web has become the leading application development platform – this is significant.

There are other marquee examples of Linux's ascendancy that are just as important. Amazon's Kindle is a set of Java applications on top of Linux. Android is also a Java VM on top of Linux. Finally, Valve is bringing video games, a long-time bastion of the Windows platform, to Linux with SteamOS.

Source:
<http://gcn.com/blogs/reality-check/2014/05/linux-wins-os-war.aspx>
 Submitted by: **Arnfried Walbrecht**

TAILS, EDWARD SNOWDEN'S LINUX DISTRO OF CHOICE, LEAVES BETA

The Amnesic Incognito Live System (Tails, for short), a Linux-based operating system purposed “to preserve your privacy and anonymity”, has left beta and is now available as version 1.0. The operating system is best known for being the operating system of choice for users who seek out anonymity on the internet, such as Edward Snowden, the former contractor for the National Security Administration. Wired reports that Snowden prefers this software, while the NSA sees this as a security threat, along with the anonymous browser Tor.

Tails is intended to be used from a bootable USB or a live disc, so any computer available can be made temporarily secure. This works by not allowing the operating system to save anything to the computer's hard disc, and force it to run entirely off the RAM memory. Many lightweight Linux distributions already allow for this, but Tails is never intended to be installed on the hard drive as to

ensure complete “amnesia.” Specific documents can be saved to the hard drive with explicit user action, but by default nothing is saved, and software is included to encrypt all external drives.

Source:

<http://www.muktware.com/2014/05/tails-edward-snowdens-linux-distro-choice-leaves-beta/26613>

Submitted by: **Arnfried Walbrecht**

WHAT WOULD YOU DO TO IMPROVE LINUX?

The Linuxsphere is a wild, wonderful and adventurous place. By its nature alone, Linux is often considered a maverick. It's been my observation that Linux users tend to explore and take more risks when using their computers. For better or worse, it appears that Linux users have been associated with the hacker elite. I've even thought of it as being the wild west of tech. I've referred to using Linux as computing without a safety net.

Source:

<http://fossforce.com/2014/04/what-would-you-do-to-improve-linux/>

Submitted by: **Peter Odigie**

LUKS: LINUX HARD DISK DATA ENCRYPTION WITH NTFS SUPPORT IN LINUX

LUKS acronym stands for Linux Unified Key Setup which is a method of disk-encryption used by the Linux Kernel, implemented with the cryptsetup package. The cryptsetup command line encrypts a volume disk on the fly using a symmetric encryption key derived from a supplied passphrase that is provided every time a partition or even a whole disk is mounted.

Source:

<http://www.tecmint.com/linux-hard-disk-encryption-using-luks/>

Submitted by: **Peter Odigie**

LINUX PROS USE TWEAK TOOLS TO CUSTOMIZE THEIR OS, AND SO CAN YOU

Where Windows has utilities, Linux has tweak tools. And whether you're a Linux pro or a recent refugee from Windows XP,

they can help you make Ubuntu 14.04 LTS “Trusty Tahr” (the latest and greatest offering from Linux distro pioneer Canonical) really start to feel like home.

Customizability has long been one of Linux's most compelling features—particularly when compared with proprietary alternatives such as Windows and OS X—but the tweak tools out there today let you refine the OS even further. And if you're making the migration to Linux on your workplace PCs, tweak tools can help ease the transition.

Source:

<http://www.pcworld.com/article/2152546/linux-pros-use-tweak-tools-to-customize-their-os-and-so-can-you.html>

Submitted by: **Arnfried Walbrecht**

KIM KOMANDO: BUY A COMPUTER FOR LESS THAN \$100

With the end of support for Windows XP last month, there are a lot of people looking to buy new computers.

If you haven't looked at computers in the last few years, you might be shocked by how much power you get for the price. The budget computers of today were the high-end computers of just a few years ago.

Of course, even a budget computer will set you back \$350 to \$500. That's not bad when you consider the speed you're getting and figure you'll have it for at least four years. But maybe you don't need to spend that much.

Source:

http://www.shreveporttimes.com/article/20140511/NEWS05/305100015/Kim-Komando-Buy-computer-less-than-100?nclink_check=1

Submitted by: **Arnfried Walbrecht**

LINUX GROUP BUILDS 64-BIT ANDROID KITKAT FOR ARM DEVELOPERS

Google is being tight-lipped about when the 64-bit version of Android will be released, but Linux development group Linaro has built a version of the open-source operating system so mobile applications can be written and

tested by manufacturers and developers rushing to catch up with Apple.

Android smartphones and tablets could be faster with 64-bit hardware and also carry more memory. Device makers are feeling pressure to catch up with Apple, which jumped ahead of the competition by putting its 64-bit A7 processor in the iPhone 5s and iPad Air. Linaro's Android builds are not full-fledged distributions of the OS, but are system builds meant for developers to write and test applications.

Source:

<http://www.pcworld.com/article/2154580/linux-group-builds-64bit-android-kitkat-for-arm-developers.html>

Submitted by: **Arnfried Walbrecht**

ENDING THE EMBEDDED LINUX PATENT WAR BEFORE IT BEGINS

The next big intellectual property battle has been forming over hardwired and programmable chips made for mobile devices that leverage Linux

code. However, the Open Invention Network has strategically deployed forces to keep Linux-powered smartphones, tablets and other computer technologies out of harm's way. Its goal is to create a patent litigation no-fly zone around embedded Linux.

The Open Invention Network, or OIN, was created in 2005 as a white hat organization to protect Linux. It has considerable financial backing from Google, IBM, NEC, Novell, Philips, Red Hat and Sony. More than 800 organizations worldwide have joined the OIN community by signing the free OIN license.

Source:

<http://www.ecommercetimes.com/story/80447.html>

Submitted by: **Arnfried Walbrecht**

HOW MUNICH SWITCHED 15,000 PCs FROM WINDOWS TO LINUX

Hirschgarten, in the west of Munich, is one of Europe's biggest beer gardens, with over 8,000 places to sit. It's a spectacular sight in summer: hundreds of benches as far as the

eye can see, trees providing some shelter from the heat, and a vast number of people relaxing and enjoying the city's famous beers.

But while 8,000 is an impressive number, it's not as impressive as 15,000. That's how many people the Munich city council has switched from Windows to Linux over the last decade. Migrating workers of Germany's third-largest city was no easy task and there were plenty of hurdles along the way, but by and large the project has been a storming success.

We've been following the progress of LiMux (Linux in Munich) for years, and now that the project is effectively complete, we decided to visit the city and talk to the man in charge of it. Read on to discover how it all started, how Microsoft tried to torpedo it, and whether other cities in the world can follow Munich's lead...

Source:

<http://opensource.com/government/14/5/how-munich-switched-15000-pcs-windows-linux>

Submitted by: **Arnfried Walbrecht**



Last month I began introducing the various concepts and terminology used by Git. This month we'll be taking it a step further, and running through a series of examples on actually using Git (in combination with GitHub; creating a custom Git server falls outside of the scope of this article). I will be assuming you either have a free account with GitHub, or else that you have found someone's repository you'd like to clone. For anyone who is interested, I created a public repository for this article, which can be found here:

<https://github.com/lswest/Command-Conquer-Examples>.

Note: If you don't have a GitHub account, and are using someone's repository, you will not necessarily be able to push to them. You can create your own copy of the repository by forking it (requires an account as well).

STEP ONE: CREATE A REPOSITORY

If you would prefer to use an existing repository (or not create a GitHub account), skip this step.

At the very top of the screen when logged into GitHub, you'll see your username, and next to that a plus sign. Click the plus to create a new repository. You can enter the following information:

Required:

- Repository Name
- Owner
- Public/Private (Private is available to only paying members)

Optional:

- Description of the repository
- Initialize with a README
- .gitignore settings
- Choosing a license for the repository.

STEP TWO: CLONE THE GIT REPOSITORY

There are some GUI Tools for Linux, but for the sake of this article, I will refer to only the actual terminal commands:

- Open a terminal.
- cd to any subfolder where you want the repository located.
- Enter:

```
git clone
https://github.com/lswest/Command-Conquer-Examples.git
```

- Depending on the size of the repository, it may take a moment. Once you're back at your prompt, you have successfully cloned it. You may now want to cd into the repository folder (in my case: cd Command-Conquer-Examples).

STEP THREE: MAKING CHANGES

For the sake of this example, I'll be editing the README.md file. Naturally, the procedure is the same with any edit:

- Open the file in your favourite editor.
- Make your changes.
- Once the file is saved, you then need the following command:

```
git add <file>
```

Note: if you want to add all files, simply use a period instead of the name of the file.

STEP FOUR: COMMIT

Once you've added the file you want, and are ready to commit them into the next update to your repository, you can run the following command:

```
git commit
```

If you run the command like this without any arguments or parameters, it will prompt you to enter a message for the commit before finishing. A couple useful options are:

- -a: Automatically adds any changes files (but ignores newly added/removed files).
- -m <msg>: Uses the message supplied for the commit message – i.e. -m "First commit – edited README.md".

STEP FIVE: PUSH

Now that you've created a

commit listing all the changes you want to take over onto the remote repository, you're ready to synchronize them. In Git this is called a push. The format of the command is:

```
git push <local> <remote>
```

By default when using Github, the local will be origin, and the remote will be master. If you're working on a more complicated repository, you'll need to check the exact branch names. In this example, the command will most likely be:

```
git push origin master
```

If you haven't configured git to store your username and password for Github, you'll be prompted for those now. As is the custom in Linux, the prompt for the password will not display any characters at all, but will indeed register your keystrokes.

STEP SIX: CHECKING THE REMOTE

Once you've pushed your changes, you'll see the newest commit listed on the repository

page. The commit message appears only on files that were affected by it. This helps to keep track of which files aren't being changed frequently.

FURTHER POSSIBILITIES

POSSIBILITY ONE: ADDING A NEW FILE

In step 3, we discussed how to edit files that already exist in the repository. First create a new file using whatever method you need, and then add it to a commit with `git add <file>`. Once you've added it, follow steps 4-5.

POSSIBILITY TWO: REMOVING A FILE

If you want to remove a file from the repository, simply deleting it from the filesystem isn't enough. You'd need to use the git command, `git rm`. Once a file is deleted with this (i.e. `git rm "Adding a file.txt"`), you can then commit the change and push it to the repository, by following the steps 4-5.

Hopefully this article has helped

to shed some light on using GitHub and Git repositories. If you have any difficulties, or want information on a specific scenario, feel free to email me with your questions. As always, I can be reached at lswest34+fcml@gmail.com. For anyone who is interested in hosting their own git servers, I would be happy to write an article on this topic too (if there is sufficient interest).



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.



EXTRA! EXTRA! READ ALL ABOUT IT!

Our glorious news reporters are now posting regular news updates to the main Full Circle site.

Click the NEWS link, in the site menu at the top of the page, and you'll see the news headlines.

Alternatively, look on the right side of any page on the site, and you'll see the five latest news posts.

Feel free to discuss the news items. It's maybe something that can spill back from the site into the magazine. **Enjoy!**



Many years ago, I was dealing with high blood pressure issues. My doctor suggested that I do something that allowed me to concentrate on something fairly useful, but rather trivial. I dealt with it by trying to do counted cross stitch. It's creative, focused, and keeps your mind occupied on what you are doing, not what is bothering you. I find myself in that position again, so I broke out the hoop and needles and started again.

In case you aren't familiar with counted cross stitch, I'll give you a gross overview of what it is. Cross stitch is a type of needlework that uses tiny 'x' patterns of thread that eventually make up a picture. The thread is called "floss" and the fabric that you use is called "aida". According to Wikipedia, aida is a special fabric that has tiny squares that have small holes at regular intervals that form the squares. This facilitates the placement of the "x" patterns that make the image. There are two types of cross stitch. One has an image printed on the aida (sort of like paint by

numbers), and the other uses totally blank aida that you count stitches from the pattern. The second is much harder than the first. Go to your favorite fabric store or craft section of your local mega-mart and you'll get the idea.

Also a while back, I started playing with creating a program that would take an image and convert it into a cross stitch pattern. One thing lead to another, and I had to shelve the program for other things. I've now dusted off the idea and started anew.

We will spend the next few articles dealing with this project. It will take a while, since some things are fairly complex and have many parts to them. Here is the "game plan":

- Create a database for the pixel colors to floss colors.
- Create a GUI using Tkinter for the application.
- Flesh out the application to do the manipulation of the image files.
- Create a PDF file that will be the ultimate pattern for the project.

- What you will learn
- Revisitation of database and XML manipulation.
 - Revisitation of Tkinter GUI programming. If you missed the previous articles on this, please refer to FCM issues 51 thru 54.
 - Image manipulation using PIL (<http://pillow.readthedocs.org/en/latest/>).
 - PDF creation using pyFPDF (<https://code.google.com/p/pyfpdf/>).

GETTING STARTED

The first thing in our list of tasks is to create the database that will hold the DMC(™) floss colors and reference them to the closest approximation to the RGB (Red, Green, Blue) values that are used in images on the computer. At the same time, the database will hold the hex value and the HSV (Hue, Saturation, Value) representations

for each floss color. It seems that HSV is the easiest way to find the "closest" representation of a color that will match the floss colors. Of course, the human eye is the ultimate decision maker. If you are not familiar with HSV color representations, there is a rather complex writeup on Wikipedia at http://en.wikipedia.org/wiki/HSL_and_HSV. It might help, but it might make things less clear.

The first thing we need is an XML file that has the DMC floss colors with a RGB conversion. The best one I found is at <http://sourceforge.net/p/kxstitch/fature-requests/9/>. The file you want is dmc.xml. Download it and put it in a folder that you will use to hold the Python code.

Now we will be using apsw (below) to do our database manipulation, which you should

```
# makedb.py
# DMC.xml to SQLite database
# For Full Circle Magazine #85
```

```
import apsw
from xml.etree import ElementTree as ET
tablename = "DMC"
```

already have and ElementTree to do the XML parsing (which is included in Python version 2.7+).

As always, we start with our imports. In this program, we have only the two. We also set the name of the table.

The next portion should be familiar if you have been reading the articles for a while. We create a function that will read the XML file, and parse it for us. We then can use the information to load the database. A snippet of the XML file is shown top right.

We are looking for the <floss> tag for each line of information. To do this, we use the .findall('floss') command. Once we have the information line, we break each tag (name, description, etc.) into separate variables to place into the database. When it comes to the <color> tag, we use the .floss.findall('color') command to get each value of Red, Green and Blue.

We start by telling the function that we will be using the global variables connection and cursor. We then set the filename of the XML file, parse the XML file, and

get started. We also use a counter variable to show that something is happening while the parsing and database inserts are going on.

Now that we have all our data, we need to create the SQL insert statement and execute it. Notice the “\” after the word VALUES in the SQL statement. That is a line-continuation character to make it easier for printing here in the magazine. We will be creating the database and table in a few moments.

```
SQL = "INSERT INTO DMC
(DMC,Description,Red,Green,Bl
ue) VALUES \
```

```
(' %s', '%s', %s, %s, %s) " %
(name, desc, red, green, blue)
```

```
cursor.execute(SQL)
```

Now, we print to the terminal window that something is going on:

```
print "Working record
{0} ".format(cntr)
```

```
cntr += 1
```

Now we create and/or open the database in the OpenDB routine (bottom right). If you've been with us when we have done database

```
<floss>
  <name>150</name>
  <description>Dusty Rose Ultra VDK</description>
  <color>
    <red>171</red>
    <green>2</green>
    <blue>73</blue>
  </color>
</floss>
```

```
def ReadXML():
    global connection
    global cursor
    fn = 'dmc.xml'
    tree = ET.parse(fn)
    root = tree.getroot()
    cntr = 0
    for floss in root.findall('floss'):
        name = floss.find('name').text
        desc = floss.find('description').text
        for colour in floss.findall('color'):
            red = colour.find('red').text
            green = colour.find('green').text
            blue = colour.find('blue').text
```

```
def OpenDB():
    global connection
    global cursor
    global ucursor
    global dbname
    connection = apsw.Connection("floss.db3")
    cursor = connection.cursor()
    ucursor = connection.cursor()
```

work before, you will notice that we are using two cursors this time. The cursor variable is used for the “normal” inserts, and later on in the select statement for the update to set the hex and HSV values. We have to use two cursors,

since if you modify a cursor in the middle of a logic statement, you lose everything with the new command. By using ‘ucursor’, we can use that for the update statements. Other than that, it is our normal OpenDB routine.

Now that the database is created and/or opened, we can set up our table (top right). Notice that the SQL statement below uses the triple quote to allow for the line to break neatly for viewing.

The EmptyTables routine (middle right) is there just to make sure that if we want to or need to run the application more than once, we start with a clean and empty table if it exists.

If we were to stop here, we would have a reasonable working database with the DMC color, color name and the RGB values associated with each. However, as I alluded to before, it is easier to pick the closest floss color by using the HSV data.

We next create the hex value from the RGB values (middle left).

The next function creates the HSV values from the RGB values. I found the algorithm on the internet. You can research it there.

Finally, we create the UpdateDB function (next page, top left). We use the SELECT * FROM DMC command and use the "standard"

```
def MakeTables():
    sql = '''CREATE TABLE IF NOT EXISTS DMC
            (pkID INTEGER PRIMARY KEY, DMC INTEGER,
             Description TEXT, Red INTEGER, Green INTEGER, Blue INTEGER,
             HEX TEXT, H INTEGER, S INTEGER, V INTEGER)'''
    cursor.execute(sql)
```

```
def rgb2hex(rgb):
    return '%02x%02x%02x' % rgb
```

```
def EmptyTables():
    sql="DELETE FROM %s" % tablename
    cursor.execute(sql)
```

cursor variable to hold the data. We then step through the returned data, and read the RGB values, and pass them to the rgb2hex function as a tuple and to the rgb2hsv function as three separate values. Once we get the return values, we use the update SQL command to match the proper record by using the primary key (pkID). As I stated before, we have to use a separate cursor for the update statement.

The last thing we do is call each of the functions in order to create the database, and, at the end, we print "Finished" so the user knows everything is done.

```
OpenDB()
MakeTables()
EmptyTables() # Just to be
safe
ReadXML()
UpdateDB()
print "Finished"
```

I named this program "MakeDB". The database should be

```
def rgb2hsv(r, g, b):
    r, g, b = r/255.0, g/255.0, b/255.0
    mx = max(r, g, b)
    mn = min(r, g, b)
    df = mx-mn
    if mx == mn:
        h = 0
    elif mx == r:
        h = (60 * ((g-b)/df) + 360) % 360
    elif mx == g:
        h = (60 * ((b-r)/df) + 120) % 360
    elif mx == b:
        h = (60 * ((r-g)/df) + 240) % 360
    if mx == 0:
        s = 0
    else:
        s = df/mx
        v = mx
    return int(round(h,0)), int(round(s*100,0)), int(round(v*100,0))
```

created in the same folder where the code and XML file are located. As always, the full code can be found at <http://pastebin.com/Zegqw3pi>.

Next time, we will work on the GUI. We use Tkinter for the GUI, so, in the meantime, you might want

to refresh your memory by looking at FCM issues 51 thru 54 where I take you through Tkinter.

Until next time, have a good month.



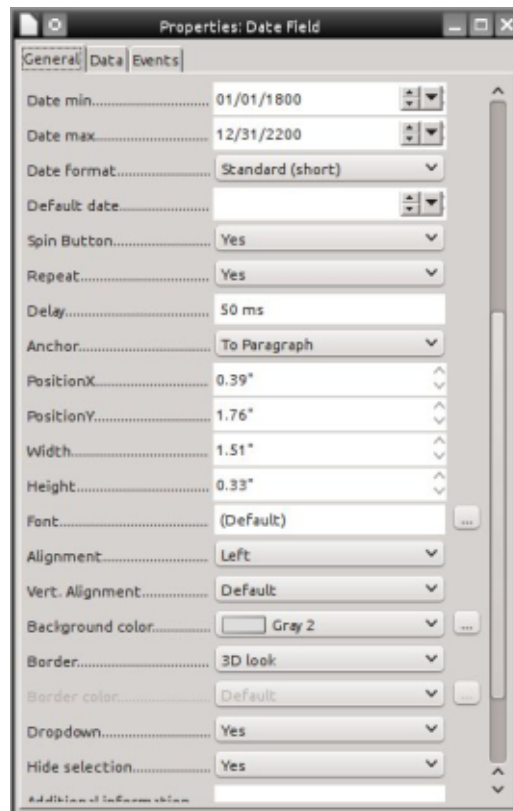
In my article in issue 83 of Full Circle Magazine, I walked you through creating a form that used a view, but I left things unfinished in the forms: I didn't do anything to format the controls to ensure data integrity. If you haven't already done so, read the article in Full Circle 83, and follow along to create the tables and forms. We will work with the "ProjectForm," and see how we can improve data integrity. I'll wait if you need to work through the How-To in issue 83.

DATE AND TIME CONTROLS

Now let's get down to work. On the "Forms" pane in the database file you created, right-click the "ProjectForm" and select "Edit". As I have said before, the wizard usually joins a label with an input box, but in the case of date and time fields, it actually pairs them up with date and time controls. Without any formatting, they just look like text input boxes, but trust me, they are date and time controls. We will need to ungroup the control from its label in order

to work with the date control. Right-click the "Due" label or date control, then select Group > Ungroup.

Double-click the date control to bring up the control's properties dialog. On the "General" tab, you find all the adjustments you can make to the control. We are interested in the ones that create adjustments for date entry.

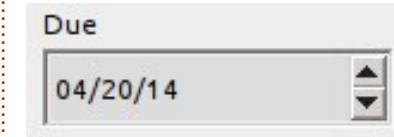


Date Min / Date Max – These two options allow you to control the minimum and maximum date values allowed in the fields. These fields have default values of 01/01/1800 for minimum and 12/31/2200 for maximum. In general applications, these defaults should work, but if you have different needs, feel free to adjust. An example is an inventory database, where no shipping invoices dated before the inception of the company is necessary. You could set the minimum date to the opening day.

Date Format – There are 12 different date formats. This option determines how the control will display the date. Scroll through the options and see which one best suits your application.

Default Date – This is a default value for the field. If you have a date that users should use most of the time, then use that date. It will speed up data entry. It can also show the user the format in which to enter dates. Unfortunately, at this time, you cannot easily set the

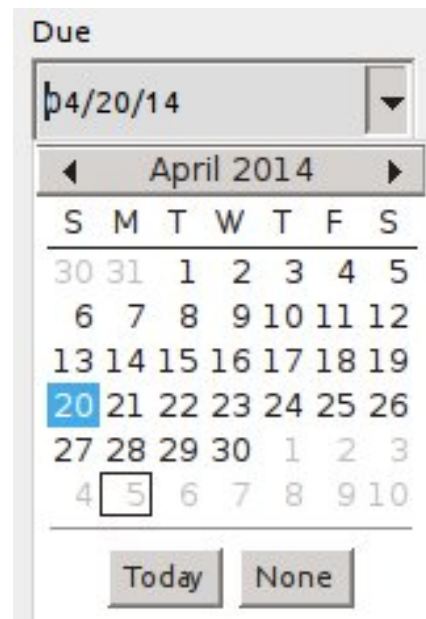
default to the (current) date the record was added.



Spin Button – Setting this option to Yes will put an up-and-down spin button on the control. The user can use this button to adjust the month, day, and year. The user clicks into the part of the date they want to adjust and makes changes by clicking on these buttons. Very handy for data entry.

Repeat – The Repeat option is related to the spin button option. It controls whether to repeat when you hold down the mouse button on the spin button. Use at your own discretion. Sometimes, it makes sense. Other times, not so much.

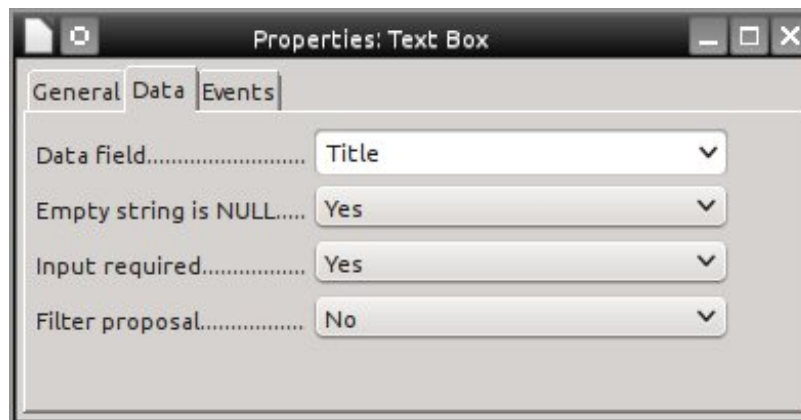
Delay – The Delay option controls the speed of the repeat option. A longer delay means you have to hold the button down longer before it will repeat. The default is "50 ms" (milliseconds).



Dropdown – Dropdown adds a listbox style arrow to the control. Clicking it displays a calendar where you can select the date you need.

NOTE: You can use the Spin Button and Dropdown options together to create a flexible control.

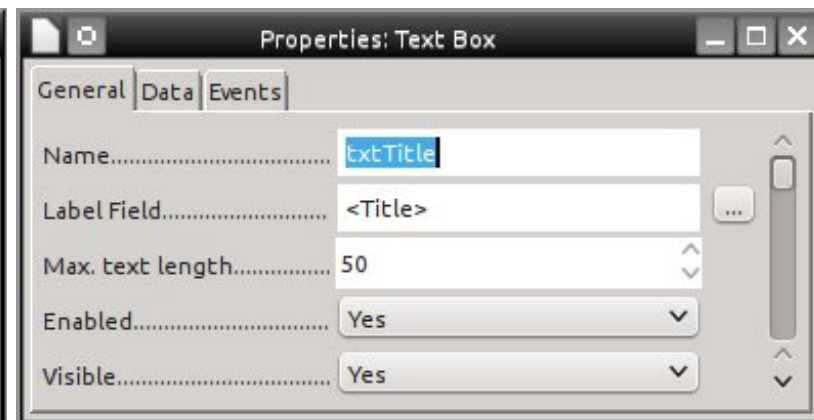
After you make your adjustments to the control, save the form and close it. Open the form and test your newly adjusted date control. Feel free to experiment with the different control options to get a better idea of how they work. The options for a time control are the same as the date control except you don't have



the Dropdown option.

MAKING THE TITLE MANDATORY

Now that we have set up the date control, let's tackle some possible issues with the "Title" field's input box. When we created the field, we set its maximum length at 50. However, as the input box stands now, we can type more than 50 characters into the box. Trying to save the record with more than 50 characters in "Title" results in an SQL insert error. Also, when we created the table, we declared "Title" as "NOT NULL." We want to make the "Title" a mandatory entry for each record. The "NOT NULL" status prevents the input box from being left empty, but it is still possible to type just a space and the form will



accept the entry as valid. So, we have a few things to fix in our form.

Open "ProjectForm" for editing. Right-clicking the "Title" label or input box, Group > Ungroup. Now, double-click the "Title" input box to bring up its properties dialog. On the "General" tab, set the value for "Max Length" to 50. This will prevent the user from entering more than 50 characters. On the "Data" tab, make sure that both "Empty String Is NULL" and "Input Required" are set to "Yes." These two settings become a redundancy check for making sure there is a valid value in the input field.

However, we still have an issue with the fact that the user can just create a "Title" with spaces. To fix this issue, we will need to use a little SQL. I have looked for a

graphical way to get the same result, but sometimes, when you work with databases, SQL is the best, if not only, choice. We will add a CHECK constraint to the "Project" table. Click on Tools > SQL. To enter the SQL, Tools > SQL... Use the following SQL command to add the constraint to the table:

```
ALTER TABLE "Project"
ADD CONSTRAINT NOSPACES_CK
CHECK (TRIM(BOTH FROM "Title")
<> '');
```

The ALTER TABLE command means we are making changes to the table, in this case the table "Project." To indicate the change we are making, we use the ADD CONSTRAINT command. NOSPACES_CK is the name of the constraint we are adding. It's not

required that you make your constraint names all caps. I just do it as a matter of convention. The last line is the CHECK constraint. Using the built-in function TRIM, it removes all the spaces from both sides of the string and checks to make sure it is not equal to an empty string. Now, if someone tries to use a string of spaces for the "Title", an SQL constraint error is generated.

You can do the same thing for the "FirstName" and "LastName" fields in the "TeamMember" table. The "Max Length" for the name fields is 25. I suggest the following SQL for adding the constraints:

```
ALTER TABLE "TeamMember"
```

```
ADD CONSTRAINT  
RNAME_NOSPACES_CK
```

```
CHECK (TRIM(BOTH FROM  
"FirstName") <> '');
```

```
ALTER TABLE "TeamMember"
```

```
ADD CONSTRAINT  
LNAME_NOSPACES_CK
```

```
CHECK (TRIM(BOTH FROM  
"LastName") <> '');
```

IMPROVING THE VIEW QUERY

The view we create for putting the names in the drop down list box in the "ProjectForm" is good, legal SQL, but there are a couple of problems. First, although we have prevented a name from being just spaces, the users can still add a legal name with spaces added to the beginning or end. Also, the names are not in any order.

Luckily, we can edit our view and make some additions to address these problems. Select "Tables" from the "Database" pane. Right-click "TeamView" and select "Edit in SQL View..." This brings up the SQL we used to create the view. Change the current SQL so it looks like the following:

```
SELECT TRIM(BOTH FROM  
"FirstName") || ' '  
|| TRIM(BOTH FROM "LastName")  
AS "Name", "ID"  
from "TeamMember" ORDER BY  
"LastName", "FirstName";
```

We already saw TRIM when we created our CHECK constraints. It removes all spaces from the beginning and end of "FirstName" and "LastName".

At the end, we tag on an ORDER BY command to sort the names by "LastName", then "FirstName." This means the names will first sort by "LastName", then if there is more than one of the same "LastName", it will sort by "FirstName."

With adjustments to the controls and a little SQL, we were able to make many improvements to our forms, making data input not only easier but more reliable. If you are interested in a better understanding of the SQL used in this How-To, I have a couple of resources for you.

My blog "What the Tech Am I Doing?" LibreOffice Base posts <http://eeperry.wordpress.com/category/libreoffice/libreoffice-base/>

W3Schools SQL Tutorial
<http://www.w3schools.com/sql/default.asp>



Elmer Perry's history of working, and programming, computers involves an Apple][E, adding some Amiga, a generous helping of DOS and Windows, a dash of Unix, and blend well with Linux and Ubuntu. He blogs at <http://eeperry.wordpress.com>



The Ubuntu Podcast covers all the latest news and issues facing Ubuntu Linux users and Free Software fans in general. The show appeals to the newest user and the oldest coder. Our discussions cover the development of Ubuntu but aren't overly technical. We are lucky enough to have some great guests on the show, telling us first hand about the latest exciting developments they are working on, in a way that we can all understand! We also talk about the Ubuntu community and what it gets up to.

The show is presented by members of the UK's Ubuntu Linux community. Because it is covered by the Ubuntu Code of Conduct it is suitable for all.

The show is broadcast live every fortnight on a Tuesday evening (British time) and is available for download the following day.

podcast.ubuntu-uk.org



HOW-TO

Written by Floris Vanderhaeghe

GRUB2 & Multibooting Pt.1

New Linux flavours emerge all the time. Ubuntu, Debian, Linux Mint, Bodhi Linux, SnowLinux, antiX, ... regularly come with new releases. Overwhelming! As if this wasn't enough, most distributions offer a further choice between several pre-installed desktops and default programs. Common desktop managers include Gnome, Cinnamon, Unity, XFCE, LXDE and Enlightenment. This wealth is a joy for open source protagonists. Still, we need practical ways to be able to enjoy and compare multiple distributions. A common approach is to run a live session (from DVD or USB stick), or to run a second operating system within some mystical 'virtual machine', but none of these ways allow you to fully assess what the behaviour, speed and stability would be when the distro is installed on and booted directly from the hard disk of your pc! Multiple installed distributions on your hard disk to choose from, would allow you to try out different candidate distributions in a 'real life' situation, and over a much longer time than a live

session.

Lately, I figured out an approach, and I was compelled to share that knowledge with other FCM readers. The answer is in customizing GRUB, the Grand Unified Bootloader.

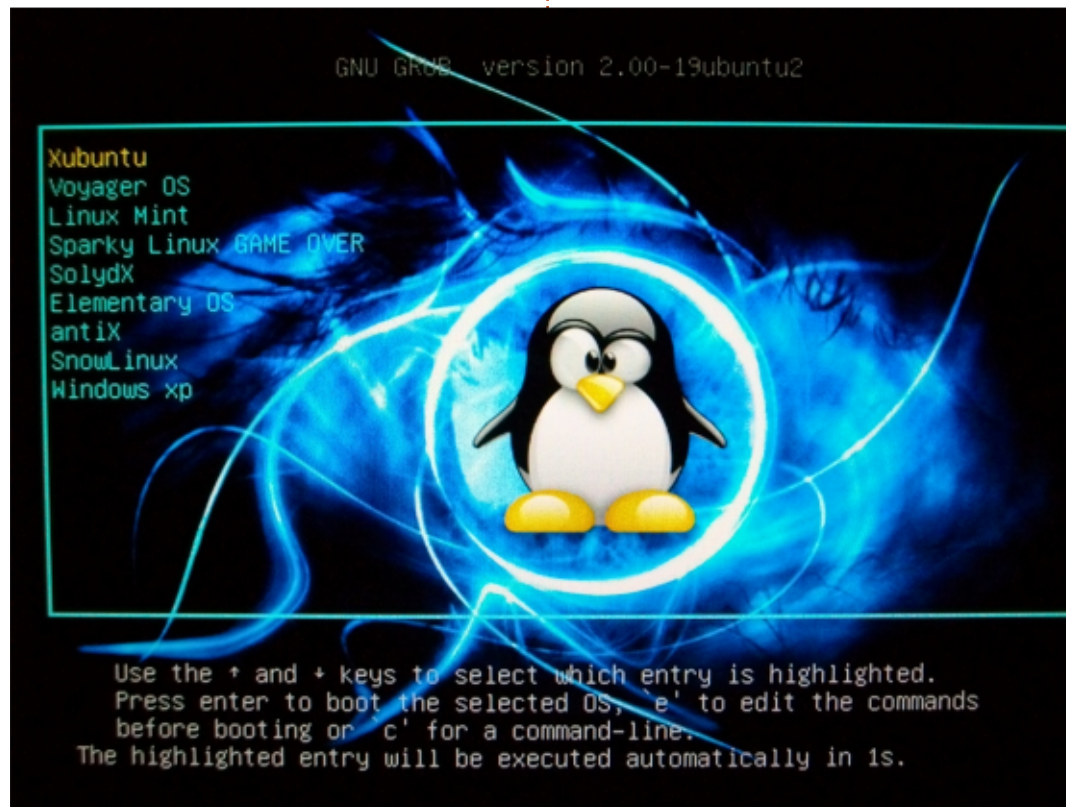
When my PC boots, the first thing that appears after BIOS is a GRUB menu in which multiple

installed operating systems are displayed to choose from (nine in my case). Many of us know about 'dual booting' Windows and one Linux distribution: it is a default option which is offered when you install Ubuntu or another Linux distribution on a Windows-only machine. After installation and restart, you will get a GRUB menu from which you can choose which operating system (OS) to boot:

Ubuntu or Windows. If you don't see the GRUB menu (e.g. you have only Ubuntu on your hard disk), press Shift during boot to unhide it. GRUB can do more for you than offering a dual-booting menu, and multibooting is what we are going to achieve in this four-parts-series! In this first part, let us discover the general purpose and behaviour of GRUB.

With GRUB, I mean version 2 (GRUB2), which is the version packaged with most Linux distributions (i.e. grub-pc package version 1.97 and later!). From Ubuntu 12.10 on, version 2.00 of grub-pc is installed.

GRUB is a bootloader, which by default will be installed on the Master Boot Record (MBR) of your hard disk. What a mouthful. The MBR is the first sector of your hard disk (512 bytes), and, as such, it is the first thing that is read from the hard disk when booting. Its contents do NOT appear within your file system(s) on the hard disk, and therefore they cannot be read by a file manager application (we



don't need to, either). It is the logical place for the bootloader, which is an in-between small operating system. The BIOS transfers its control of the PC to the bootloader when booting from the hard disk. The bootloader is essentially responsible for correctly transferring the control of your machine to an operating system. Bootloaders other than GRUB exist as well, like LiLo for example.

A bootloader like GRUB contains pointers to one or more OSs on your hard disk, as each OS is installed on a separate partition of the hard disk (or another hard disk). Usually, when more than one OS is installed, GRUB offers a menu from which an OS can be selected at startup. If you don't interfere, its default selection is booted after 10 seconds. Furthermore, GRUB is able to keep track of several installed versions of the kernel in a Linux distribution (something which dual-booters will be familiar with). Each installed kernel (within one and the same OS) will appear as a separate line in the GRUB menu (or in a submenu). In this way, it is still possible to "fall back" to a previous kernel version in case things don't work as expected with

a newly installed kernel. Of course, GRUB does not store all of its information within the small MBR sector: on installation it has been configured to look for needed modules and configuration settings in the `/boot/grub` folder on a Linux partition.

So far we have a general idea of what GRUB is responsible for. However, starting right away by installing Linux distributions A, B, C and D on partitions 1, 2, 3 and 4 will lead to trouble in the longer term. Let's see why.

Each time you install a new distribution, the GRUB that is already present on the MBR is overwritten because the new distribution re-installs GRUB from its own grub-pc package. This new GRUB is told to look for its configuration file `grub.cfg` in the `/boot/grub` directory of the newly installed Linux distro. No problem so far, because this configuration file is filled with pointers to all operating systems present on your hard disk. So you will eventually get an up-to-date GRUB menu after you installed your last Linux distribution (let's assume that is distro D). Let's also suppose you have done some final tuning of this

GRUB configuration to make the GRUB menu look awesome (e.g. with a nice background, beautiful text colours, default OS preset, etc.). Will things work as you would like? Yes – for now. However, the `grub.cfg` file that GRUB (on the MBR) relies on, is controlled solely by distribution D (which you installed on partition 4). Will a kernel update in distribution C be reflected in the GRUB menu? No, because distribution D is unaware of that kernel update. What happens when the `grub-pc` package is updated in distro A, B or C – or when you replace A with a new distro E? The MBR will be overwritten with GRUB... from distro A (or E), B or C (which does not refer to `/boot/grub/grub.cfg` on partition 4 but to the one on partition 1, 2 or 3). So... goodbye nice background and other tunings that were present in the old grub configuration file! What if you replace distro D and overwrite GRUB? Same thing.

To conclude, it is only by actively updating the `grub.cfg` from within distro D - by running the 'sudo update-grub' command - that kernel or OS changes elsewhere on your hard disk will be detected and reflected in the GRUB menu. So

that makes you very dependent on distro D. Not an ideal situation when you like to change your Linux installations every once in a while.

My purpose with this article series is to give you a better understanding of how GRUB works, and to demonstrate a more advanced option than a GRUB installation governed by one of the Linux distributions: we will achieve an independent bootloader on the MBR. By that I mean a GRUB installation which will not be affected by your replacements of one Linux distribution by another, and over which you have control yourself!

Next time, we will dive into some internals of GRUB!



Floris Vanderhaeghe became a Linux fan through Ubuntu 10.10. You can email him at tux7546@gmail.com.



HOW-TO

Written by Nicholas Kopakakis

Blender Part 11b

The first thing we need to do in order to create something, is good reference of that something. Now that we want to create the star wars intro scene, we need a little google search. First, I found a video that is just what I want to create:

<https://www.youtube.com/watch?v=UKRIUiyF0N4>

We need additional information so I continue searching for something useful. I end up on this site

<http://www.theforce.net/fanfilms/postproduction/crawl/opening.asp>

This looks like what I need. Let's download the star wars logo from this site.



Also from the site, download the free font that we will use to create the body of the text

(SWCrawlBody).

For now, the scene that we want to create is the star wars logo movement and the scrolling text (from 0:08 to 1:32 of the youtube video).

Tip: This is not a project where we will need extra super-detailed references. All we need is a video that we are going to create "as is," and some information about the elements that they have used to create the scene. If we had to create a model, let's say an airplane for example, we would have searched for a lot more photos finding out how the airplane looks from different angles and a lot of technical information such as how long is the cockpit's size or the width of the wings.

As you may notice there are some differences between the text on the video and the text on the site (on video there is no "Episode IV a new hope"). We want to create an animation just like the video, so I suggest to download it. We will need it to count for example the

duration of the crawl or the duration of the appearance of the star wars logo.

So, I downloaded the video from youtube, put it in kdenlive (<http://www.kdenlive.org/>), and measured that the star wars logo appeared for approximately 8 seconds before disappearing in the distance, and 1 minute and 16 seconds is the duration of the crawling text appearance. Because the crawling text appeared on-screen after the disappearance of the star wars logo, then the entire animation will have 8 seconds + 1 minute and 16 seconds duration. Meaning 1 minute and 24 seconds. Converting that to frames, we have: 84 seconds multiplied with 25 frames, is giving 2100 frames total. That's the exact duration of our animation.

Now we are ready to start a new blender project and save the file as starWarsIntro.blend or something like that.

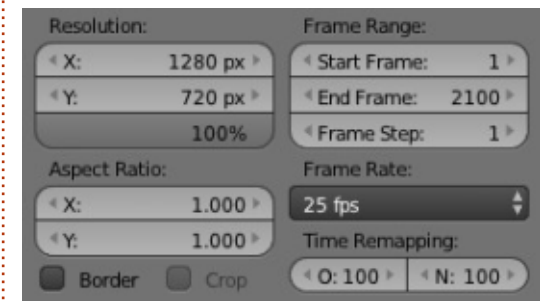
Delete the starting cube from the scene.

Now, adjust the project render settings.

Press the render icon, and, under the Resolution settings, set the width (X) to 1280 and height (Y) to 720. Also set the percentage scale to 100%.



Under the Frame Range, set End Frame to 2100



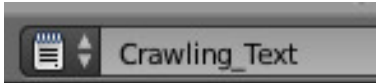
Press Numpad-7 to have a top orthographic view on your 3d view window. Now click and drag to the left, the upper right corner of the 3d view window to create another window.

HOWTO - BLENDER



Change the new window to text editor.

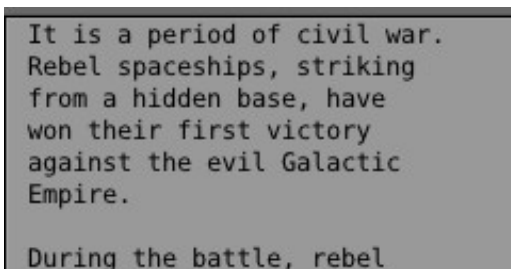
Now, press the new button to create a new text document in blender. Name it `Crawling_Text` as per the image below:



Write down the text from the image that we found on www.theforce.net website, or copy paste it to text editor from another source.

Tip: You can also write the text directly to a text object with Shift-A → Text to add a new text object but you cannot copy paste into this object or use the Tab key to index the words of your text with this method. That's why I prefer to create a text file with the text editor.

At the end you must have something like this



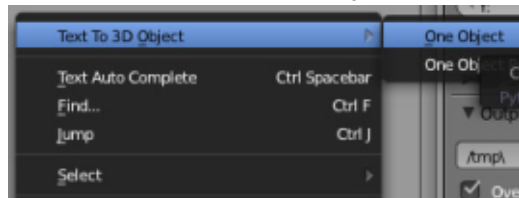
Try to make your text identical to the crawling text on the video (the first line ends with the word "war", the second line with the word "striking", etc.)

Save the project.

Now, it's time to make this text to 3d object.

At the header of the text editor, select Edit→Text To 3D Object→One Object.

Your text appeared in the 3d view window as a 3d object.



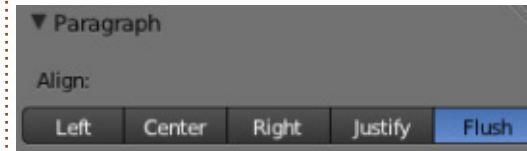
Some more adjustments to our text. Go to the properties window and press the F icon:



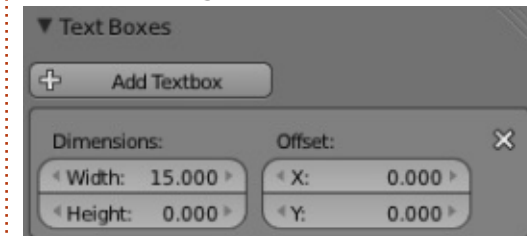
Under Font, navigate and select the font that we downloaded earlier from the www.theforce.net site.



Under Paragraph, select Flush as we want the word to fill the entire line



Nothing happened. Let's adjust one more thing. Under the text Boxes, change the Width to 15.

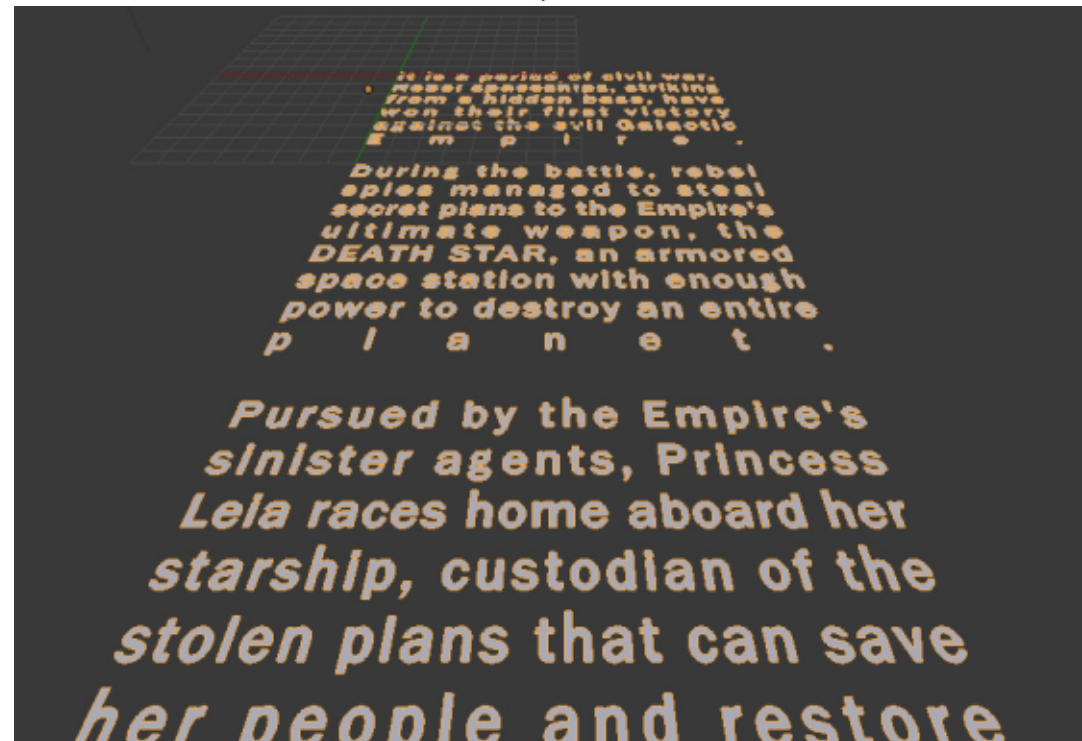


Here is what we have so far in our 3d view window (Perspective view and camera position and rotation has applied)

to be continued...



Nicholas lives and works in Greece. He has worked for a post-production house for several years and migrated to Ubuntu because "it renders faster." You can email him at: blender5d@gmail.com





HOW-TO

Written by Mark Crutch

Inkscape - Part 24

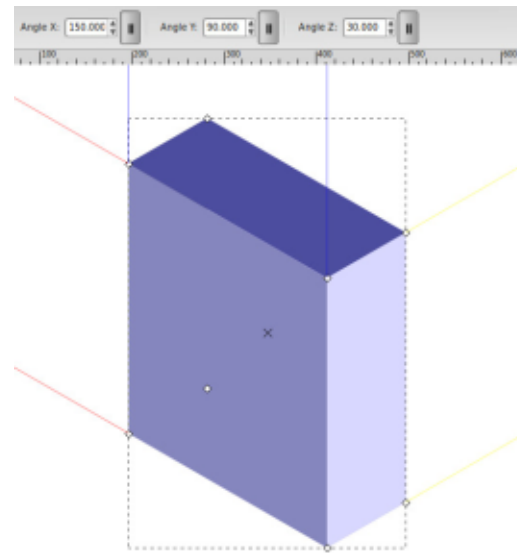
In this article I'll be looking at Inkscape's 3D Box tool. The first thing to note about the 3D Box tool is that it doesn't create 3D boxes. What it actually does is draw and manipulate six closed paths to produce a 2D representation of a 3D box – Inkscape is, after all, only a 2D drawing program, not a 3D solid CAD application. This is also the only fake-3D drawing primitive that Inkscape offers, so, if you want 3D shapes other than boxes, you're on your own. With those limitations in mind, let's take a closer look to see how the 3D box tool works, and how you can use it as the basis for more complex representations of 3D objects.



You can activate the 3D Box tool using the icon on the toolbar, or by pressing X or Shift-F4. The normal 2D rectangle tool can be activated by pressing F4, so you might like to think of this tool as drawing a rectangle that's been shifted into the third dimension. Depending on where you draw, and how much you've tinkered with the 3D Box

tool before, it's easy to create a box that's too big or too distorted, making it hard to understand the tool. So before you start to draw, let's set some sensible defaults on the tool control bar.

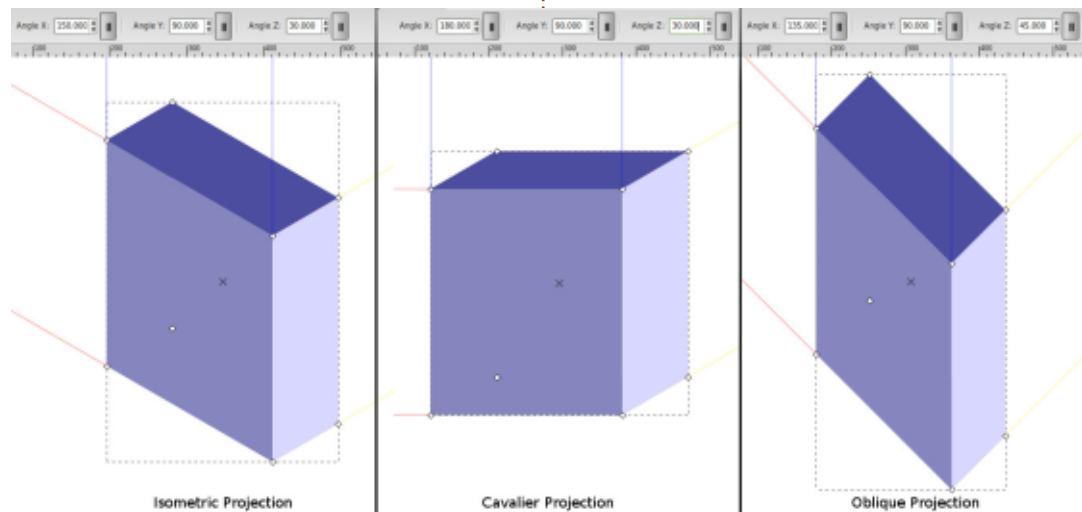
The options for this tool are deceptively simple – but that's only because a lot of the complexity takes place via handles on the canvas instead. There are three pairs of controls, consisting of a spinbox and a button for each of the X, Y and Z axes of the pseudo-3D world that your box will live in. Start by clicking all three buttons into their “on” state. This sets an infinite vanishing point for each axis, ensuring that the edges of your box are all parallel. Enter values of 150, 90 and 30 into the spinboxes – as usual there are context menus with some sensible options, though 150 isn't one of the defaults. Now click and drag on the canvas to draw your first 3D box. Make it fairly large so that it's easy to distinguish the small handles on the box when you release the mouse button.



There are a few things to note about the box you've drawn: there are red, blue and yellow

construction lines which project from the edges along the X, Y and Z axes of the 3D world; there are eight small handles, one at each corner of the cuboid (one will appear to be in the middle of the box – it's just the handle for the obscured corner showing through); there's a small X-shaped handle at the center; finally, the box is probably made up of shades of blue.

Because we've set the axes to have an infinite vanishing point, the projection lines form parallel pairs. The angle of each axis is measured in degrees, in an anti-clockwise direction where 0° points



to the right. Compare the angles you typed in earlier with the projection lines – by using 30° for the Y-axis, and 150° for the X-axis (calculated as 180° minus 30°), your boxes form a classic isometric projection. Try setting the X angle to 180° for a Cavalier projection, or set X=135° and Y=45° for an Oblique projection.

The square handles are used to adjust the size of your box. Four of them can be used to modify two dimensions (X and Z) at once, whilst the other four let you change only the third dimension (the depth along the Y-axis). Holding Shift will toggle the handles, so that a handle which usually controls the Y dimension instead lets you change X and Z – and vice versa. When moving an X/Z handle, try dragging it a little way along one of the projection lines, then holding Ctrl. This constrains the movement to one direction only, should you wish to change the X dimension without altering the height, for example.

The X-shaped handle can be used to move the whole box, and once again Ctrl can be pressed to constrain the movement to one axis. Usually it's best to move 3D boxes using this handle, rather

than using the Selector tool – it doesn't make a lot of difference for these projections, because all the axes have infinite vanishing points, but once we turn off one of those toggle buttons the behaviour of the two methods diverges significantly.

With the projection lines and handles dealt with, it's time to talk about color. The default behaviour for the 3D Box tool is to draw your boxes in shades of blue. You can try setting the fill color before you draw, but you'll still get shades of blue. And if you set the fill color afterwards – well, you'll get the color you chose, but not as shades. Rather, all six faces of your box will be set to the same fill color and you'll end up with something that looks more like a flat, distorted hexagon than a 3D box.

The secret to coloring your box is to understand what it's made of. The SVG format doesn't allow for 3D primitives, and certainly doesn't know anything about 3D boxes. What Inkscape creates is actually an illusion, made up of six paths that are grouped together. These paths carry some Inkscape-specific attributes so that the program knows to treat them differently to

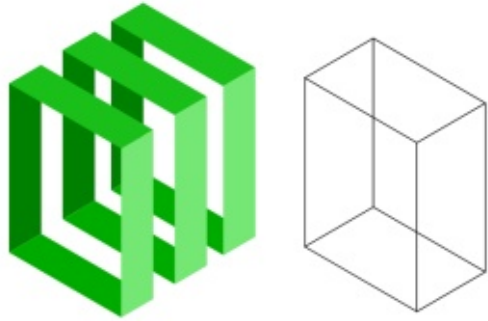
a normal group of paths, but they're just a group of paths nevertheless. That's why all six sides take on the same fill when you click on a color swatch – they're just behaving the same way that any group of paths does in that situation.

The key to changing the color is therefore to enter the group and modify each path individually. Whilst double-clicking with the Select tool usually enters a group, in this case it just switches to the 3D Box tool, so instead you need to right-click and select “Enter Group #g3116” from the bottom of the context menu (the exact title will vary based on the ID of the group). Alternatively you can click on the box using the Selector tool, then press Ctrl-Enter, or Ctrl-Click to select one of the visible faces without actually entering the group.

Once you've entered the group, I suggest starting by moving the faces apart from each other. This will give you a better idea of how the box is constructed. Use Shift with the arrow keys to move by a precise amount, if you want to be able to move the faces back to reconstruct the box later.



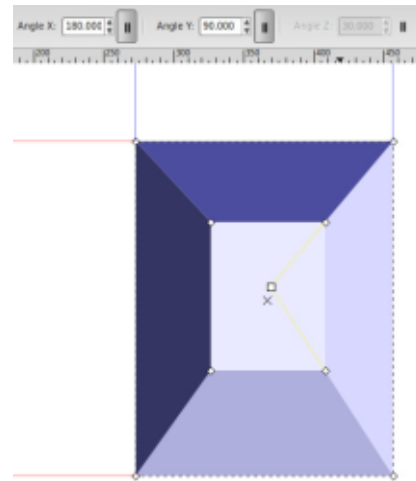
While inside the group you can also change the color of each face individually – or even delete some of the faces entirely. Once you're done, exit the group by double-clicking on the canvas background with the Selector tool, picking “Go to parent” from the context menu, or using the Layers pop-up in the status bar. You may be surprised to find that Inkscape still considers your object to be a 3D box even if it's exploded, a different color, or with faces missing. This can be useful when you want a box with no lid, or you just want to give each face a stroke but no fill to create a simple wireframe.



The projections we've used so far are okay for certain types of technical drawings or 3D pixel art, but the 3D Box tool can also create images with perspective. Let's start with a simple 1-point perspective.

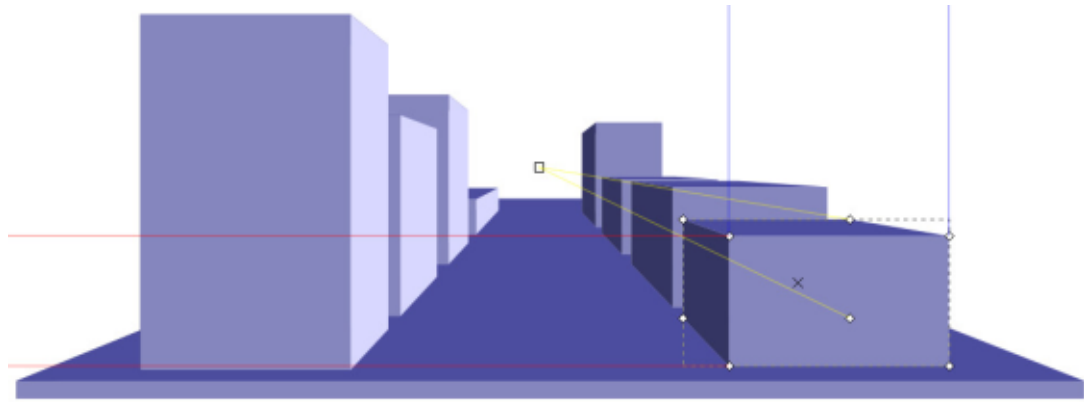
We'll begin by removing the front left face of our box (medium blue, with the default colors). You should now be able to see into the box to the pale back wall. Set your X-axis angle to 180° , as we did for the Cavalier projection, but this time click on the parallel lines button for the Z-axis so it's deactivated, to give the Z-axis a vanishing point. You'll notice that the Z-axis angle is no longer editable, and instead the two yellow projection lines now converge to a square handle on the canvas. That's the vanishing point (VP) – try dragging it around to see the effect. Put it somewhere near the center of your box to give the

appearance of looking down a corridor. You may need to adjust the corner handles for the best effect.



With the 3D Box tool still selected try drawing some more boxes. Notice that they all share the same VP. By moving them using the X-shaped handle they will continue to share their VP, but if you move a box using the Selector tool, the VP will also be moved, breaking the link between that box and any others.

Using this shared VP behaviour, it's easy to create a few boxes with 1-point perspective to form the basis of a street view. You'll need to adjust the corner handles for each box, but remember to hold Shift if they won't move in the right direction.



Returning to a single box, let's try some 2-point perspective. Draw a new box if you need to, but this time, toggle the parallel lines buttons for both the X-axis and the Z-axis. Leave the Y-axis with an infinite VP at an angle of 90° .

Now both the red and yellow projection lines converge into square handles at the VPs. Drag the handles around – for a good effect put the red one a little above the box and far to the left, and the yellow one at a similar height, but far to the right. Now drag the box around by the X-handle, and note what happens as it moves above the VPs, or close to either of them (hold Shift to move it along the Z-axis).

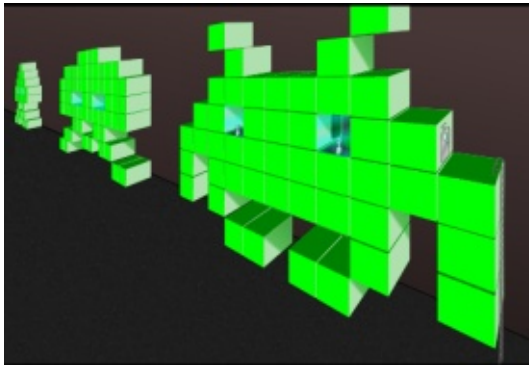
Once again, any other boxes you draw will share the same vanishing points. Try starting with a small box near the red VP, then repeatedly

duplicate it (Ctrl-D) and move the duplicates to the right along the projection lines (hold Ctrl while dragging the X-handle). Duplicate again and move up, along the Y-axis (start moving up, then press and hold Ctrl). Very quickly you can build up a wall of boxes in this way or, as I did in a recent comic strip, leave a few gaps to create a 3D "pixel" image.

As you may have guessed, you can also turn off the parallel button on the Y-axis to give you 3-point perspective. This is not as frequently used as the previous options, but can be great if you want to draw huge, imposing towers and monoliths.

If you do move a box using the Selector tool, you may wish to reconnect it to the VPs of your other boxes. Simply Shift-click on

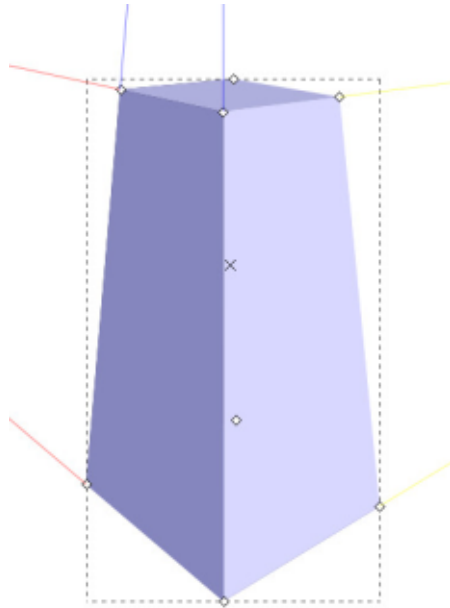
HOWTO - INKSCAPE



multiple boxes when the 3D Box tool is active and you'll be able to see all the VP handles at once. Drag the handle from one box onto the handle of another to snap them together. Be warned, this operation pays no heed to the color of the handles you're connecting, so you can easily end up joining the X-axis of one box to the Z-axis of another. Quite what the result will be in any artistic or mathematical sense, I don't know – but feel free to experiment if you really want to. If you decide that you need to separate any joined VPs, then ensure that only one box is selected and hold Shift as you drag the VPs around. Alternatively, use the Selector tool to move the whole box.

The 3D Box tool may seem quite specific and even restricted in what it can do, and if you take it on face value then that's probably true.

But by setting a stroke with no fill (and you don't even need to enter the group to do that), you can create a wireframe box that makes it much easier to draw in perspective. So don't think of it as a tool for drawing boxes, think of it as a tool for drawing construction lines. Once you've got the basic frames for your street, wall or tower, the rest is down to your imagination.



Mark's Inkscape created webcomic, 'Monsters, Inked' is now available to buy as a book from <http://www.peppertop.com/shop/>

PYTHON SPECIAL EDITIONS:



<http://fullcirclemagazine.org/issue-py01/>



<http://fullcirclemagazine.org/issue-py02/>



<http://fullcirclemagazine.org/python-special-edition-issue-three/>



<http://fullcirclemagazine.org/python-special-edition-volume-four/>



<http://fullcirclemagazine.org/python-special-edition-volume-five/>



<http://fullcirclemagazine.org/python-special-edition-volume-six/>



HOW-TO

Written by Ronnie Tucker

Arduino

Last month I reviewed the Arduino Starter Kit. From this month onwards, I'll be documenting my trials and tribulations with the Arduino and its programming language.

SUNFOUNDER 37-IN-1

I've also purchased the Sunfounder 37 sensor modules kit (via Amazon). It's a great little kit

for the money (albeit that some 'sensors' are just normal parts mounted on a little circuit board), but the documentation for it is almost non-existent. It doesn't even describe the pinouts on them. You get a badly translated (pidgin English) PDF. Here's an example from the start of the PDF:

"Let us into a variety of interactive electronic world. . ."



```
int Led=13;
int Shock=3;
int val;

void setup()
{
    pinMode(Led, OUTPUT);
    pinMode(Shock, INPUT);
}

void loop()
{
    val=digitalRead(Shock);
    if(val==HIGH)
    {
        digitalWrite(Led, LOW);
    }
    else
    {
        digitalWrite(Led, HIGH);
    }
}
```

"With Arduino digital pin connected to terminal 12 can be, even on the USB data Throughout the test circuit is complete, DS18B20 test results we will use the serial port are displayed on the computer screen. We will test the compiled code downloaded to the board, open the serial port can know that we are What kind of temperatures in the Well, did not talk much, we first look at the test code bar"

It pretty much gives the same code for every module. Basically, if the sensor is activated then illuminate the (onboard) LED on pin 13.

So, some code. I've been through most of them and run the basic code (shown above) to show it going on/off:

ONWARDS!

HOWTO - ARDUINO

What I've been doing is linking a couple together to practice coding. In this example I hooked up the avoidance sensor and a buzzer to light the pin 13 LED and buzz when I put my hand near the sensor. The red wire goes from + on the sensor to the live, yellow from Gnd to negative and the green wire goes from the S pin on the sensor to digital pin 3. The buzzer also goes to negative (yellow wire) with the green wire going to digital pin 2. The sensor has a fourth pin (with EN beside it) but I've no idea what

it does.

Although, I've no idea what the fourth pin on the avoidance sensor does. It's cryptically labeled with EN beside it.

My code is shown below.

Yes, the if statement does seem a bit backwards, but that's the only way I could get the sensor to work properly. It took a good few hours of fiddling to get this far with no documentation, four badly labelled

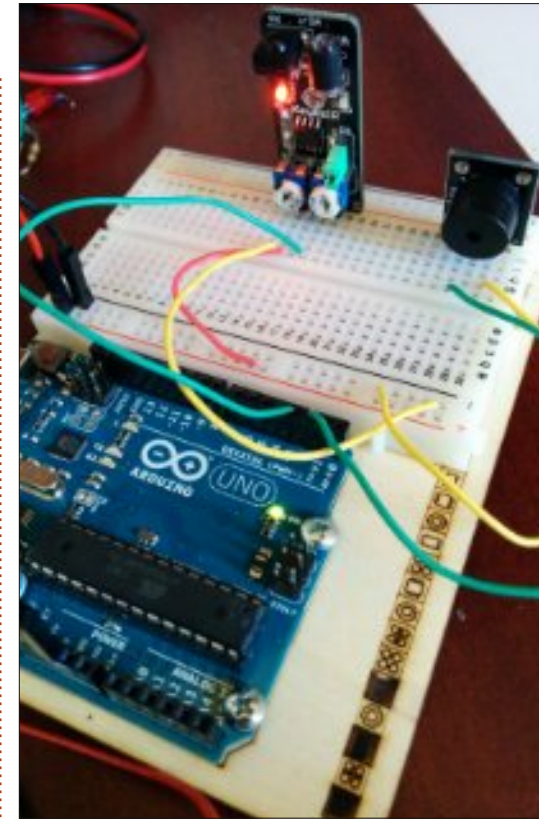
pins, two adjuster screws and an old fashioned jumper block! From what I can tell, the jumper block seems to make the sensor look behind itself rather than out in front as it's shown in the photo (and in the code), with the two screws setting avoid distance and sensitivity. I think.

As I always say: it's all fun and games until it starts getting hard...

```
int Led = 13;           // define LED
int Avoid = 3;          // define the avoid sensor pin
int Buzzer = 2;         // pin 2 for the buzzer
int val=0;              // define a variable

void setup ()
{
    pinMode (Led, OUTPUT);           // define LED as an output
    pinMode (Avoid, INPUT);          // input for avoidance sensor
    pinMode (Buzzer, OUTPUT);        // set buzzer pin as output
}

void loop ()
{
    val = digitalRead(Avoid);        // read avoidance sensor and store result in val
    if (val == HIGH)                 // if something is NOT near sensor then go high
    {
        digitalWrite (Led, LOW);     // turn LED off
        digitalWrite (Buzzer, LOW);  // turn buzzer off
    }
    else
    {
        digitalWrite (Led, HIGH);    // turn LED on
        digitalWrite (Buzzer, HIGH); // turn buzzer on
    }
}
```



Ronnie is the founder and (still!) editor of Full Circle. He's a part-time arts and crafts sort of guy, and now an Arduino tinkerer.





HOW-TO

Written by Ronnie Tucker

Write For Full Circle Magazine

GUIDELINES

The single rule for an article is that **it must somehow be linked to Ubuntu or one of the many derivatives of Ubuntu (Kubuntu, Xubuntu, Lubuntu, etc).**

RULES

• There is no word limit for articles, but be advised that long articles may be split across several issues.

• For advice, please refer to the **Official Full Circle Style Guide:**
<http://url.fullcirclemagazine.org/75d471>

• Write your article in whichever software you choose, I would recommend LibreOffice, but most importantly - **PLEASE SPELL AND GRAMMAR CHECK IT!**

• In your article, please indicate where you would like a particular image to be placed by indicating the image name in a new paragraph or by embedding the image in the ODT (Open Office)

document.

• Images should be JPG, no wider than 800 pixels, and use low compression.

• Do not use tables or any type of **bold** or *italic* formatting.

If you are writing a review, please follow these guidelines :

When you are ready to submit your article please email it to:
articles@fullcirclemagazine.org

TRANSLATIONS

If you would like to translate Full Circle into your native language please send an email to ronnie@fullcirclemagazine.org and we will either put you in touch with an existing team, or give you access to the raw text to translate from. With a completed PDF, you will be able to upload your file to the main Full Circle site.

REVIEWS

GAMES/APPLICATIONS

When reviewing games/applications please state clearly:

- title of the game
- who makes the game
- is it free, or a paid download?
- where to get it from (give download/homepage URL)
- is it Linux native, or did you use Wine?
- your marks out of five
- a summary with positive and negative points

HARDWARE

When reviewing hardware please state clearly:

- make and model of the hardware
- what category would you put this hardware into?
- any glitches that you may have had while using the hardware?
- easy to get the hardware working in Linux?
- did you have to use Windows drivers?
- marks out of five
- a summary with positive and negative points

You don't need to be an expert to write an article - write about the games, applications and hardware that you use every day.





Online
BACKUP

Secure
SYNC

Easy
SHARING

Whether you need to access a document you have stored on a remote server, synchronize data between a Mac, Windows or Linux device, share important business documents with your clients, or just rest easy knowing all of your data is safely, securely, and automatically backed up - SpiderOak's free online backup, online sync and online sharing solution can handle all your needs!

SpiderOak offers a different approach to online backup by combining a suite of services into one consolidated tool - free online backup, synchronization, sharing, remote access, and storage. This difference is further measured in our zero-knowledge privacy policy - the first one ever employed in this setting. Our flexible design allows you to handle data from any operating system (Mac, Windows and Linux) or location (external drives, network volumes, USB keys, etc...) using just one centralized account.

- ➔ Access all your data in one de-duplicated location
- ➔ Configurable multi-platform synchronization
- ➔ Preserve all historical versions & deleted files
- ➔ Share folders instantly in web ShareRooms w / RSS
- ➔ Retrieve files from any internet-connected device
- ➔ Comprehensive 'zero-knowledge' data encryption
- ➔ 2 GBs Free / \$10 per 100 GBs / Unlimited devices

<https://spideroak.com>

Download mobile clients
for **iOS & Android**

JOIN SPIDEROAK NOW
Get 2 Free GBs

Get 25% off any SpiderOak package
with the code: **FullcirclemagFans**



ASK THE NEW GUY

Written by Copil Yáñez

Hi, everyone! Welcome back to Ask the New Guy!

If you have a simple question, contact me at copil.yanez@gmail.com.

Today's question is:

Q: I hear people talk about 'compiling from source.' What does that mean?

A: Seriously? You're going there?

Fine.

Compiling from source is one of those things that separates the wheat from the chaff. The folding chair from the Aeron. Justin Bieber from Chris Martin (because, apparently, Chris is a hardcore Ubuntu user, and Justin still runs Windows 95 *snort*). There's your life before you learned to compile from source, and everything that comes after.

Let me be very clear about this. First, there's no going back. Once you learn to compile from source,

anything is possible.

Second, contrary to everything you read online, learning to compile from source will not, I repeat, NOT, make you more attractive. Also, you won't be able to move things with your mind, or breath underwater. Invisibility? Not even a real thing.

Compiling from source opens up new dimensions in Ubuntuing (a word I'm still desperate to make happen). But it won't give you superhuman powers. Are we all clear on the limitations?

Okay, good. Let's learn what compiling from source is. Then you can try it at home, under controlled conditions, and, preferably, with a loved one standing by to tell you how normal you still are when it's all over.

First, the dreaded explanation of what 'compiling from source' even means. This is where I go out and read page after mind-numbing page that explains what something means, all so that I can come back

here and distill the information into a completely unintelligible, grossly unintelligent, marginally usable definition that's almost always riddled with factual errors, and may even cast aspersions at your mother's honor just for laughs. Just part of my charm.

Let's start with an explanation of an executable.

What?! I thought we were talking about source code and compiling! And now you throw this curveball?

Well, actually, I'm throwing you a tarball, but we're getting ahead of ourselves. Look, just calm down and trust that I'll lead you out of this dark cave we entered without a map. I know this place like the back of my hand, and there's a backpack with light sticks and sandwiches right over here. . .er. . .over here. No, wait, it's right here. . .you know what? We all gon die!

Back to executables. An executable is simply a program your Ubuntu operating system can

understand and use. If you're using the graphical interface, and click on LibreOffice, the program that runs is an executable.

Executables use binary code, a sort of magic and sorcery that humans can't read unless they're Neo from The Matrix. It's just a bunch of ones and zeros that look like gibberish to you and me, but Ubuntu gets all hot and bothered about.

How is it possible to create programs if humans can't read, write, or work directly with the gibberish of an executable file? That question, my friend, leads us back to source code.

See? I told you we'd get out of that cave! Granted, we're now knee deep in quicksand and sinking fast. But no worries. I have something in my backpack just for this. My backpack that's over there, at the edge of the sandpit. Just out of arm's reach. Dammit, we're gonna die again!

Relax, source code is just a list

of instructions that a human can read, write, and interact with. Those instructions are written using a programming language like C++, and they can be turned into an executable (more on how in a minute).

So writing a program in Linux goes something like this: The spark of creation > programmer fires up C++ > programmer writes a bunch of code she can read and understand > that code gets turned into an executable somehow > end user runs the executable.

Why wouldn't programmers go all the way through to turning their source code into an executable? Are programmers inherently lazy?

Hm, not gonna touch that question with a ten foot pole.

Actually, there are plenty of reasons why a programmer wouldn't want to go all the way through to creating the executable. First, it takes time. And the bigger and more complex the program, the longer it takes to create the executable. More importantly, each distro is slightly different. The programmer would

have to create a slightly different executable for each distro. So even with smaller programs, a few minutes for each executable needed for different distros would mean a programmer could spend hours just on executables. That's a lot of time that could be better spent creating new programs or consuming vast amounts of bacon.

What's more, having an executable that can't be played with, modified, updated or patched by others kind of defeats the purpose of open source programs. See, you can go from source code to an executable, but not the other way around. So let's say you have

the perfect tweak to an open source program, but all you have is the executable. You can't turn that back into source code to modify the code and add that killer fix.

So a programmer will release the source code and now everyone can play with it. When someone needs to run it on his particular distro, he simply compiles the final executable.

Hey! We're out of the quicksand! We're finally back to the original question, what does it mean to "compile" from source code?

When you compile the source code, you're using a program, known as a compiler, to take the program you created in one language, and turn it into a program in a second language. So it's like either using a computer version of the Rosetta Stone, or mixing ground chicken parts into chicken feed; the recursiveness of it all has me a little confused.

But you get the idea. Compiling is the process of taking those human-readable instructions (the source code) and turning it into machine-readable instructions (sometimes called the object code) that make up the executable.

Let's update that flowchart for writing a program. Now, it looks something like this: Case of Red Bull delivered by Amazon Prime > spark of caffeine-fueled creation > programmer fires up C++ > programmer writes source code > programmer compiles source code > end user runs the executable.

Now here's the exciting part: where it says "programmer compiles source code," that can be you! No waiting around for a pesky programmer to do it for you. No begging your Ubuntu-hipster

```
function DisplayMailboxes(caption, list) {
    if (list == "")
        return "";
    if ((list == user.PrimaryMailbox) && (
        return "";
    var mboxlist = new String(list).split("
    var row_bar = "<TR><TD COLSPAN=2 CLASS=
    var row_space = "<TR CLASS=row_normal>
    var text = "<TABLE CELLSPACING=0 CELLP
        "<TD COLSPAN=2 CLASS=row_top><TABLE (
        SPC(8,20) + "</TD><TD WIDTH=100% CLAS
        caption + "</TD></TR></TABLE></TD></T
```

friend, the one who goes on about how he was totally into Ubuntu when it was just a little gleam in Mark Shuttleworth's dreamy eyes.

You CAN DO IT! YES, YOU!

If you've been using Ubuntu for a long time, and have never had the need to compile from source, I'm not surprised. Before Ubuntu was popular (so, for the ten minutes before anyone heard about it), not all programs were compiled for our distro. But as time went on, and more people used Ubuntu, fewer programs needed to be compiled because someone had already done it. So instead of downloading a file that contained the source code, and then compiling it for your operating system, you simply downloaded the executable. Better yet, you just went into the Software Center and installed it from there.

But every now and again, there's a program out there that you want or need, that's written for Linux, and that hasn't been made available through the Software Center. Do you just throw up your hands and pout? Maybe. And if your significant other is a

power user, that might actually do the trick. But if not, you can handle it yourself. Here's how.

First, you need the source code, the instructions in C++ or other programming language that, when compiled, will become the executable program. These come in many flavors, but they usually look something like this:

```
FILENAME.tar.gz
```

A tar file is the bundle of files that make up the source code. Because the tar bundle can get pretty big, you sometimes use gzip to compress them into a file ending in .gz.

Once you download the tar file, it's time to extract it. You can do that at the command line by typing:

```
tar -xzf FILENAME.tar.gz
```

You may have already guessed that to unpack a tar file, you use the tar command. You'll see some letters after the dash (xzf). These are called switches, and they tell Ubuntu to do certain things to the file that follows. In this case, we ask the tar command to unzip (z) and then extract (x) the following tar

file (f).

Once the file is extracted, you need to configure it. Unlike compiling, configuring involves finding out what kind of computer you're running, and then "configuring" the source code to work with it. Assuming you're running as root administrator, here's the command to configure the source code:

```
./configure
```

Now that you have the source code, and you've configured it for your particular computer, you would think the last step of compiling the source code would use a command like, say, oh, I dunno, COMPILE!! Yeah, not so much.

Instead, you'll use the Make command, like this:

```
make
```

Simple, right? This command looks at the configured source code and compiles it into an executable that will run on your computer. Technically, this is the compile step in the expression "compile from source."

Woohooo, you've compiled from source! Is that it? Maybe. If you're one of those totally evolved beings who would climb to the top of Everest and not take any pictures, then I guess you are. The rest of us need something to remember this momentous occasion by. Like, maybe, by using the actual program we just compiled. Our last step, then, is to install the executable. To install the program you just compiled, we use the Make command again like this:

```
make install
```

This puts all the compiled files where they're supposed to go. Think of it as a really good usher at the theater, who makes sure you're not seated on top of someone else, which happens a lot more than most people realize.

Now you can run the program! The program YOU compiled all by yourself! From the command line, no less!

Go, YOU!

Now listen to me, LISTEN TO ME! I suggest you take a little

break, have a cookie, and just relax for the rest of the day. I can't count the number of times I've heard about a new user learning to compile from source, and then going out and signing up for MMA lessons because she thought she was Ronda Rousey!

Like most things you can do in Ubuntu, compiling from source can get pretty complicated, pretty fast. Start with small programs, preferably ones with decent wikis or README files that explain any tricky bits.

Besides allowing you to use programs that might not be easily available for your system, learning to compile gives you the freedom to work directly with the program. It's like giving you the keys to a really cool car and letting you go under the hood to add a supercharger.

There's something about seeing the actual code used to create the executable that gives you a sense of power! I have, like, this overwhelming urge to run outside and toss the caber!

HULK SMASH! I just compiled from source!

Think I'll join a Fight Club.

Good luck, and happy Ubuntuing!



Copil is an Aztec name that roughly translates to "you need my heart for what again?" His love of women's shoes is chronicled at yaconfidential.blogspot.com. You can also watch him embarrass himself on Twitter (@copil).



After reviewing PinguyOS last month, I tried out several distributions over the period of a few days: Fedora 20, Ultimate Edition, Elementary OS, Zorin OS 8.1, Linux Deepin, and Kubuntu 14.04 (beta). Many years ago, in a galaxy really, really close, I used to run Red Hat Linux (and later Fedora 1 through about 8). Then I joined a resistance with friends Paul Nijjar and Daniel Allen to create WCLP, Working Centre Linux Project (long defunct but still available), a distribution for the computer recycling project we volunteered for. WCLP was based on Debian GNU/Linux. Since that time I've tended to favor Debian-flavoured distributions.

Of all the distributions I tried after PinguyOS, one stood out as a clear winner for me, and something of a surprise (to me). I've tried Kubuntu several times over the years and there was always something that made me switch away rather quickly. That was actually the experience with most of the previously mentioned Linux distributions, except for Kubuntu

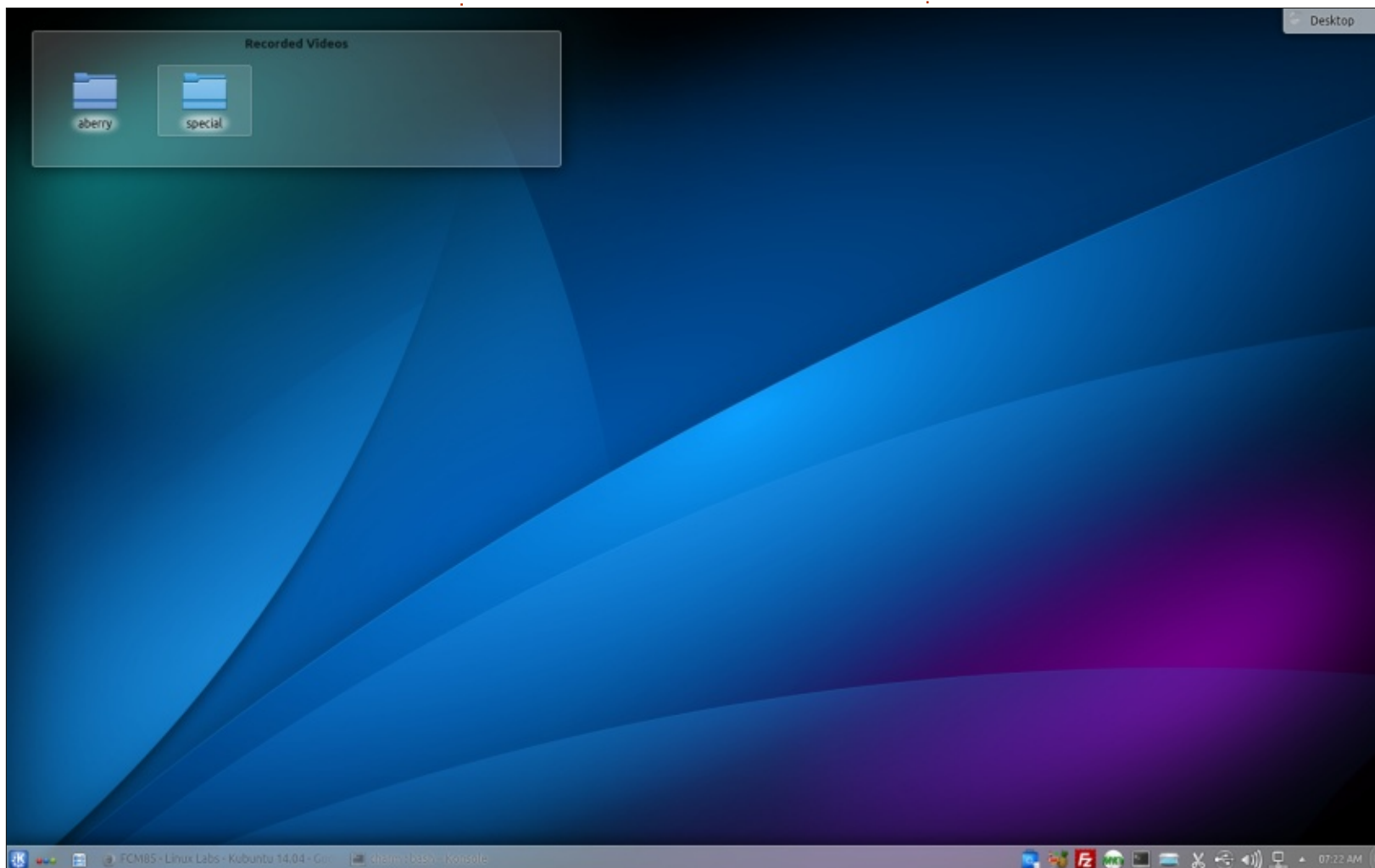
14.04.

Kubuntu 14.04 surprised me by being more stable than any other previous version I've tried (even though I was trying the beta at the time). In the past, stable

applications under Kubuntu have been one of my biggest pet peeves - I found applications tended to crash far too much under Kubuntu. It's been more than 20 days now and I've only had 1 or 2 (application) crashes during that

time.

Firefox is the default web browser for 14.04. Currently I have version 29.0 installed. Firefox has been my go-to web browser under Windows (at work I sometimes



have to RDP to my Windows box from my Xubuntu system) and Linux for a long time now.

I'm still not a huge fan of search-style graphical user interfaces, and the way Kubuntu nests applications seems like it might be difficult for someone with bad arthritis, but I've gotten used to it. One graphical change I liked was the work done on Muon software manager; it looks a little more like a software store. It still doesn't have quite the polish of Ubuntu's software center, but it's a little prettier and a tiny bit more functional.

I installed Handbrake and MakeMKV almost immediately because they're applications I use regularly (Handbrake at least, my Blu-ray collection is about 8 strong). Both worked without issue and neither have crashed since I installed them. I think I mentioned last month loving the speed difference between this AMD A8-5600k and a Core 2 Duo. Ripping the same DVD was a matter of 13 minutes versus 50 on the Core 2 Duo using the same software. Buying new technology makes a huge difference.

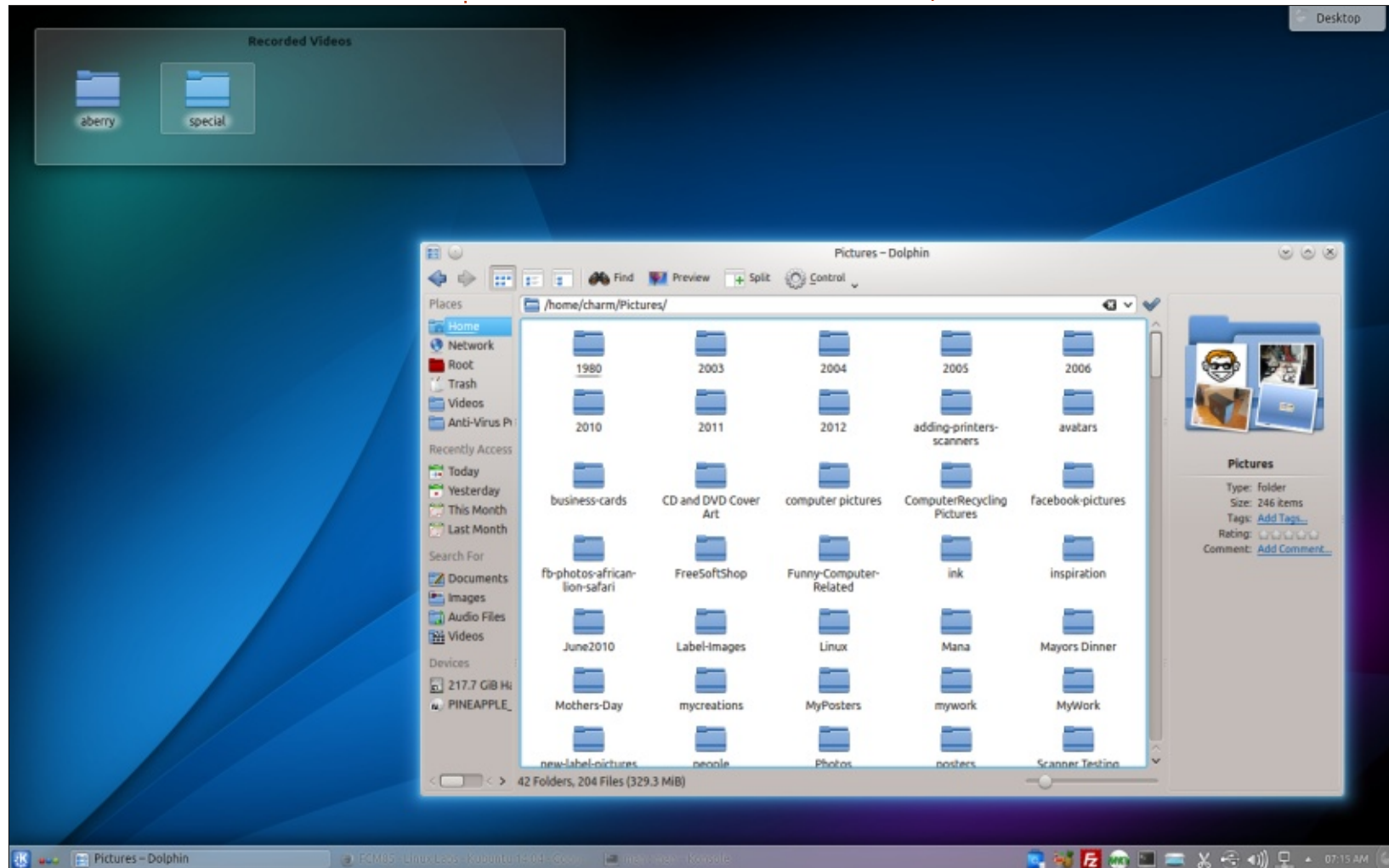
One of the tasks I needed to do was scan a dozen documents. I found the default scanning program, Skanlite, to be laggy with my HP Scanjet 2200c, so I installed Simple Scan. Although it takes a little bit for my Scanjet to warm up each time, Simple Scan seems to work much better. I also needed to send the scanned documents with

a password. I discovered that Ark, the default graphical compression application, doesn't seem to be able to zip up a document with a password. File Roller, the default application for compression on a number of Gnome-based distributions, has a well implemented password protection function. I ended up using the

terminal to zip my documents with password:

```
zip -P passwordhere  
zipfilename.zip file1.ext  
file2.ext file3.ext
```

I was sending the zipped files to a Windows user who had no issues opening up the zip files.



I also needed to transfer the documents to another computer. I could have used FileZilla, but I tend to use this when I have large, or many, documents to transfer. I transferred some of my media files using FileZilla without issues. For these documents I used Dolphin, pressing CTRL+L to get the editable location bar and typing smb://myserver/ to connect to myserver via Samba. I also love Dolphin's split-pane functionality and the fact that I can drag any folder to the Places bar on the left for quick access. As I explored Dolphin further, I found I was able

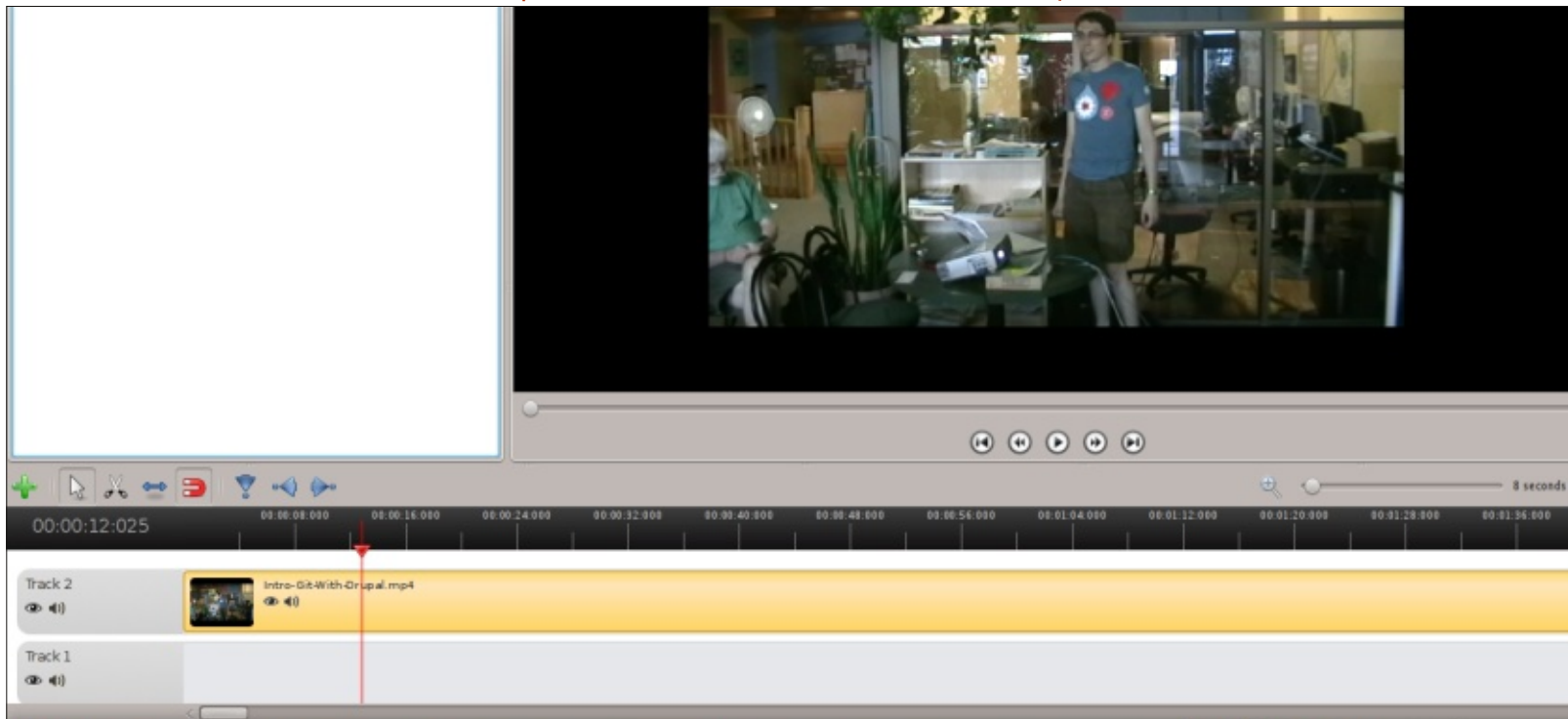
to tailor it more to my taste through the control menu, permanently enabling the editable location bar and showing the full path rather than just the current directory. Also worth checking out are the services you can download under Control > Services > Download New Services. While I'm sure a lot of this functionality existed before in Dolphin, the stability issues I had with applications in Kubuntu before really prevented me from exploring Kubuntu for very long.

Another job I did over the past

month was putting together some video previously imported from our DV camera. The DV file was roughly 45 minutes long. I installed Openshot and Blender, imported the digital video file and created the video complete with Blender-rendered animated titles. Rendering titles in Blender took longer than I would have liked, I really understand why video production staff like working on 8 core Xeon-based systems. But in the end I was able to produce a video for family that had a lot more polish than the raw DV and not 1 crash in the entire process.

One thing I noticed when I was using Linux Mint (before the PinguyOS review last month) was that when I clicked on any of the rounded rectangular sections on page 2 of Full Circle Magazine issues, I wasn't being taken to the article page - this works in Kubuntu 14.04.

Overall I've been really pleased with Kubuntu 14.04. Like every operating system I've ever used, there are minor annoyances (Ark and passwords), but the freedom that GNU software gives us means there are lots of applications we can switch to if we don't like a piece of software. For now I've found my desktop distribution and I'm surprised and happy to say it's Kubuntu.



Charles McColm is the author of *Instant XBMC*, and the project manager of a not-for-profit computer reuse project. When not building PCs, removing malware, encouraging people to use Linux, and hosting local Ubuntu hours, Charles blogs at <http://www.charlesmccolm.com/>.



MY STORY

Written by Jim Dyer

My Start With Ubuntu

I don't recall just when I started using Ubuntu, but it was 18-24 months ago. I was looking for a Windows XP alternative, and ran across several references to Linux, and specifically to Ubuntu. Did a bit of online reading and a book or two. Finally I jumped in and set up a desktop PC to dual-boot Ubuntu and Windows XP. I ran like that for some months until I managed to screw up the dual-boot setup. Then followed a little agonizing and a decision to wipe the SSD and install only Ubuntu 12.04 LTS. A bit of a steep learning curve at times, but the books helped a lot and I found a series of articles ("30 Days with Ubuntu Linux") written by Tony Bradley for PC World magazine. Some of those gave technical help, and others just boosted my confidence. All said and done, not too bad a learning curve and I had some fun at the same time.

I use Firefox, Chrome, Thunderbird, and LibreOffice, frequently. Also, a number of others as needed. My one on-going frustration is with scanning. It is often a hassle – when I need to

scan – to get the software to “find” my Canon USB scanner. I started with Simple Scan, but now use Xsane – which does a better job. Printing to my Samsung laser printer has never been an issue, nor has Internet.

My computer:

- Built from a kit via tigerdirect.com

- CPU: AMD Athlon 64 bit
- RAM: 4GB DDR2
- SSD: 60 GB - Ubuntu 12.04 LTS lives here
- External USB HDD: 500 GB - Ubuntu and previous Win XP Documents, formatted FAT32
- Display: HP 17 inch

Now that Windows XP is

officially unsupported, today I took another older desktop PC, wiped XP, and installed Ubuntu 14.04 LTS - more fun to come. The machine is older and slower, but up to the task.





REVIEW

Written by Robin Catling

Ubuntu 14.04

It seems like a good day to revisit the hallowed halls of Ubuntu; the latest Long Term Support (LTS) release of Canonical's still-popular Linux distribution, Ubuntu 14.04, "Trusty Tahr" was released April 17.

And having played with the sole beta build of Ubuntu 14.04 LTS, it turns out that Trusty (a Tahr being an African goat) is a much stronger release than recent lacklustre versions.

Two things struck me when I loaded up 14.04 in a virtual machine; the Unity launcher just works (it didn't before in VirtualBox), and the whole desktop look is so much cleaner and crisper. A lot of new code has gone into the desktop; you can see it in the Unity Greeter, and it's carried through to the lock screen.

Given there's no fall-back mode, no Gnome2 and no plan B, the Unity desktop has to work. On everything. I expect much of the work has been done for those running 14.04 on HiDPI screens (such as a Retina Macbook), to the

benefit of the rest of us.

The Unity launcher, Dash, Window Controls theme, icons and text all scale proportionally, showing up third-party applications, many of which look fuzzy and distorted. The UI scale can be set from System Settings, Displays. 14.04's 'bling' extends to anti-aliased corners, borderless windows, and a new Unity Control Centre, all largely enabled by the move from Compiz to GTK3+.

Consequently, the 14.04 selection of wallpapers is quite stunning to show things off; Jelly

Fish, Water Web, Reflections, Forever Shady, Berries and a new default wallpaper are among the artistic highlights.

Locally Integrated Menus (LIM) should please many; now Canonical has backed off its 'my way or the highway' stance; for the first time since the introduction of the Unity desktop launcher, Ubuntu is offering users a choice of where application menus appear.

By default, the menus continue to appear in the top panel, but you can now choose to return them to the application's own menubar

position; tick the Local Menus option located in System Settings, Appearance, Behaviour.

There's another tacit climbdown over usability. Anywhere in Windows or Mac, and on most Linux desktops, one click on a launcher icon will minimise a running application off the current workspace. Unity now joins in with Minimise on Click.

Admittedly it is an 'unsupported' extra that requires you install the CCSM application via the Software Centre, before you can set Minimise on Click at the Unity section, 'Launcher' tab.

14.04 runs a 3.13 Linux kernel. The current stable builds of Firefox, Thunderbird and LibreOffice are included by default, along with old GNOME favourites – Rhythmbox, Gedit and the Terminal; patched - or 'unpatched' - restore full application menus. Nautilus has also been reset so that 'type ahead' searching works in Ubuntu again.



With Xorg 1.15 and Mesa 10.1; Unity 7.1 sits atop GTK3+ 3.10.7 and the now-familiar stable of default applications:

Nautilus 3.10.1
Firefox 28
Thunderbird 24.4.0
LibreOffice 4.2.3
Ubuntu Software Center 13.10
Gedit 3.10.4
Totem 3.10.1
Rhythmbox 3.0.1
Empathy 3.8.6
Transmission 2.82
Deja Dup Backup Tool 29.5
Shotwell 0.18.0
Brasero 3.10.0

TRIM is enabled by default for Intel and Samsung solid-state drives (SSDs), Nvidia Optimus graphics support is also improved if that's your hardware.

What is missing may be more telling: Mir, the new graphics stack being developed by Canonical to replace X Windows is still nowhere ready. There's no touch-enabled desktop for tablets, either, although we may see the first of the long-awaited Ubuntu smartphones toward the back end of the year, along with a touch interface that could be in 14.10.

But pet-hates persist: Unity lenses. What I still don't like is the ghastly shopping lens in Unity Dash. This is still opt-out rather than opt-in; it's blunt, artless, child-unfriendly, and an invasion of privacy. All in the name of trying to cream a little commission from a tiny percentage of on-line purchases.

I try to search for an application, and two-thirds of my screen is taken up with random junk from Amazon trying to sell stuff. And I mean random. I search on "up" (software "up"dater) and my screen is filled with shopping suggestions, some of which are not age appropriate.

So the first thing I need to do is disable online search in the Dash.

Simple enough to do using System Settings, Security and Privacy, Search: "When Searching in the Dash - include online search results" ON/OFF. But that takes out ALL online results, so there goes my news and weather. I suppose that's what browsers are for.

Disabling the Shopping Lens alone is possible but remains a technical challenge because Canonical doesn't want you to do this. Canonical is a commercial enterprise. Fine. If you don't like that, use a different distro. I do.

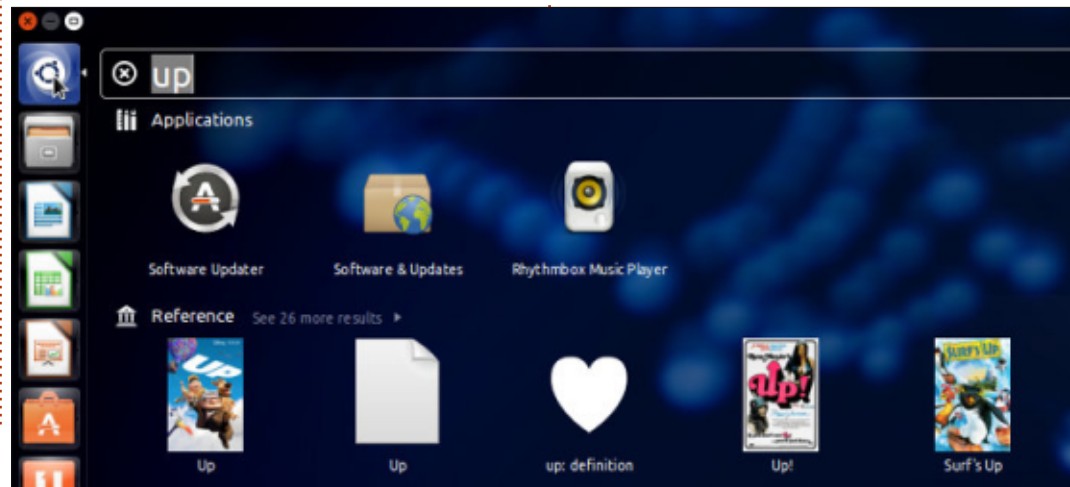
If you are a regular Ubuntu user, then, no question, it's worth upgrading to 14.04. The finessing of Unity is worth it on its own. This LTS gets security updates and

selected application updates for the next five years.

If you're not an Ubuntu regular, is it enough to make you switch (or in my case, return)? No. Canonical's commercial imperatives are still at odds with the Open Source ethos. Driving Canonical's mobile ambitions appears to have been at the expense of the desktop - even though all of that work will eventually benefit the desktop; but for how long? It's taken a long time to get Unity to a state of maturity, just as the pundits predict the death of the desktop OS.

Looking at 14.04, I still have more questions than answers; is the conventional PC desktop dying? How much effort should Canonical expend on a desktop operating system? Can Canonical make any inroads on other form factors? TVs may still be open territory, but surely Android (Linux-based) and iOS between them have the mobile space locked down?

Much as I like its highs and loathe its lows, I suspect I'll continue to run Debian or LinuxMint. You may decide otherwise.





WHAT IS

Written by Oscar Rivera

I first heard about Bitcoin a few years ago and had no idea what it was. Then, about six months ago my brother began trying to talk me into buying some Bitcoin and other types of cryptocurrency. Before embarking on such a task, I did a bit of research to make sure that I wouldn't be throwing my money away. In order to acquire other (alternative) cryptocurrencies, I first had to buy Bitcoin. Over the last couple of months, we've been looking at Bitcoin as an alternative currency here at Full Circle Magazine. Bitcoin is not the only form of cryptocurrency, but it is the oldest and most popular.

I must warn you; Bitcoin, like most other cryptocurrencies, is highly volatile, and its value climbs and drops faster than a rollercoaster. If you've been keeping up with our past articles on cryptocurrencies and are contemplating using Bitcoin (or other cryptocurrencies), I recommend you do what I did and research the specific coin that interests you. There's a good website that has charts from various exchanges – many of these



charts are put together by speculators who have been at it for a while and are pretty well informed on the subject: <https://www.tradingview.com/> Here's a couple of charts that paint two completely different scenarios: the first chart (6 month chart) shows the value of Bitcoin dropping; however the second chart (1 year chart) shows the value of Bitcoin rising. So, as you can see, there are many sides to this coin.

So, let's look at a handful of alternative cryptocurrencies. Please do keep in mind that the currencies that I'm presenting in

this article are but a tip of a very large and constantly growing iceberg. Not a day goes by without some news on how cryptocurrencies will help globalize the world's economy, while at the same time bringing third world countries into the new millennium, or, on the other hand, how cryptocurrencies will bring doom and poverty to every corner of the world. The truth is that only time will tell, and, whether we embrace them or not, cryptocurrencies will most likely stick around for quite a while.

Alt-coins

LITECOIN (LTC)

Having bought Bitcoin, I then proceeded to use my Bitcoin to buy Litecoin which is sort of like a cousin of Bitcoin. Litecoin is based on an open source protocol that is decentralized. There are some differences between the two, but that could be the topic of another article. For now, let's just say that I repeated all of the steps outlined before to establish an offline Litecoin wallet. To get your Litecoin offline wallet set up, you need to head over to: <https://litecoin.org/> which has some excellent instructions for installation on Ubuntu. To make the exchange from Bitcoin to Litecoin, I used <https://btc-e.com/> which is basically an online market exchange where you can trade different types of cryptocurrencies. For some reason, BTC-e doesn't really have much, if any, in the form of instructions. Then again, the assumption is made that if you're trying to exchange Bitcoins for Litecoins, you probably already know how the system works.



WHAT IS

Bitcoin has been called an alternative currency, and Litecoin was originally intended to improve upon Bitcoin, but in many ways Litecoin is very similar to Bitcoin. Currently, you can buy one Bitcoin for approximately \$431.79, but that cost changes every few minutes. On the other hand, one Litecoin goes for \$10.36 as of May 7 2014. Two great sources for prices and charts are:

<http://bitcoincharts.com/> and <https://www.tradingview.com/>.

DOGECOIN (DOGE)

Dogecoin can be considered a newcomer on the cryptocurrency block. Dogecoin was first introduced on December 8, 2013, and it's already become one of the most traded (by volume) cryptocurrencies. The idea behind

Dogecoin is that it is used as an Internet tipping currency, and that it's used by the common man, sort of like a de-mystified Bitcoin. For example, let's say there's a video that you like on youtube.com, and, in addition to hitting the "Like" button, you could also send a tip to the artist with Dogecoin. So far, Dogecoin has already made a name for itself by participating in some noteworthy causes. For example, the Dogecoin community raised \$30,000 to send the Jamaican Bobsled team to the Sochi Winter Olympics.[1] Most recently, the Dogecoin community once again raised money for another sport, this time for NASCAR driver Josh Wise and his Dogecoin painted (and sponsored) race car. Josh Wise finished 20th at the 2014 Aaron's 499 NASCAR Sprint Cup Race at Talladega



Superspeedway.[2]

RIPPLE (XRP)

A third type of cryptocurrency that I've been buying is Ripple, also known as XRP. It is not so easy to find information about XRP at the moment because of how new it is. Since you're a "Full Circle Magazine" reader, you can be one of the first to buy XRP. Unlike Bitcoin and Litecoin, XRP is not intended to be an alternative form of currency; instead it is sort of a bridge currency which can be used when there's no direct exchange available between two different currencies in different parts of the world. Ripple's primary design focus is to be used as a currency exchange and distributed payment system. You can set-up your account for XRP at

<https://ripple.com/client/#/register>. Just make sure to read all of the directions carefully. In the past you could only get a Ripple wallet through the Ripple Labs online client, but as of May 2014 it is now possible to download and keep an offline (also known as "cold") wallet stored in your computer.

Unlike other cryptocurrencies,

XRP cannot be mined; all of the XRP that will ever exist has already been created. An easy way to get paid in XRP is by joining the "World Community Grid" and using "BOINC" with your PC. Back in FCM issue #79, in our "How-To" section, we showed you how to setup and use "BOINC." By setting up and running "BOINC" you are joining the "World Community Grid" which is the world's largest non-profit computing grid. In FCM #79, we explained how "BOINC adds up the processing power of thousands of computers... to help process incredible amounts of data required to accomplish various scientific research projects." Our "How-to – Use BOINC" from FCM #79 does a great job at showing you how to set-up "BOINC" and at explaining what it is. If you go through the trouble of setting up "BOINC," you will not only be helping the world be a better place, but you will also be getting paid in XRP for doing so. The exact amount that you get for donating your computing resources varies according to how powerful your CPU is and how much of your processing power you're willing to donate. In my setup, I have "BOINC" using about 20% of my processing power for 12 hours a

WHAT IS

day and it uses no more than 5 GB of disk space at any given time. For such a setup, I usually get between 2-8 XRP per day; which isn't much – if you consider that, at the time, one XRP costs about \$0.025 USD. In a way, setting up and running “BOINC” is sort of like mining for Bitcoin, except that you're contributing your resources for much greater causes and still getting paid in XRP. You can also get free XRP in the form of XRP Giveaways. Because XRP is such a new cryptocurrency, there are currently lots of XRP Giveaways all around the world. I've been using “SnapSwap” to buy XRP, which has the potential to give you up to 2,000 XRP for buying through them. Upon an initial deposit of \$10.00 USD, you automatically get 500 XRP, and, with every transaction thereafter, you get 3 XRP for every \$1.00USD you deposit onto your “Ripple” account. XRP is said to be one of the most secure and reliable of all cryptocurrencies available, which makes it one of the most appealing to look into.

NAMECOIN (NMC)

Namecoin is another very

interesting type of cryptocurrency. What separates Namecoin from the rest is that, in addition to being a form of currency, it also serves as an alternative decentralized DNS. One of its primary purposes is to fight against censorship. There is a great video on YouTube about free speech and how Namecoin fights against censorship. Check it out here:

<http://www.youtube.com/watch?v=lkKsiTKxe-s>. The video is short but to the point. If you'd like more information about Namecoin, I suggest you go to the official website at: http://dot-bit.org/Main_Page.

The process of setting up an offline Namecoin wallet is the same as with Bitcoin or Litecoin. The download required to set up your offline wallet can be found at



the Namecoin official website. Much like Litecoin, you can trade your Bitcoins for Namecoins at BTCe or other cryptocurrency exchanges. You can also mine for Namecoin and it's even possible to use the same device to mine for Namecoin AND Bitcoin at the same time! If online privacy, security and censorship are important to you, maybe you should look into Namecoin and the Free Speech movement associated with it. You might just find what you're looking for.

CONCLUSION

The cryptocurrencies mentioned here are not the only ones available. In fact, that's just the tip of the iceberg. There's an overabundance of different types of cryptocurrencies available.

Chances are that there's a cryptocurrency out there that's just right for you. As with anything new, my advice is that you research as much as you can about whichever cryptocurrency you're thinking of buying. I've seen the values of cryptocurrencies skyrocket, plunge and stabilize more than once in the six months that I've been trading these currencies. I encourage you to check it out; it just might be your thing.

Sources:

- 1 - <http://www.theguardian.com/technology/2014/jan/20/jamaican-bobsled-team-raises-dogecoin-winter-olympics>
- 2 - <http://motherboard.vice.com/read/talladega-shibe-josh-wises-highlights-in-the-dogecar>



Oscar graduated from CSUN, is a Music Director/Teacher, beta tester, Wikipedia editor, and Ubuntu Forums contributor. You can contact him via: www.gplus.to/7bluehand or email: www.7bluehand@gmail.com



CONVERSION

I am retired but spent my last 10 working years as a developer in Windows using dotnet etc. After retirement, I no longer had access to a free library of software, and I realised that my existing system would soon become obsolete. My son introduced me to Ubuntu and I used it alongside Windows. I learned Java and enjoyed programming in this language. About this time I also splurged out on Windows 7, but found an annoying bug; it would not let my machine sleep whereas Ubuntu had no such problem.

I found myself using Ubuntu more often, and I enjoyed testing various Linux flavours. There were some problems with scanning and printing but there was a whole community willing to help me out. When Ubuntu adopted Unity and changed Nautilus it was not to my liking so I tried Linux Mint with Nemo. It was perfect for me, and now I have removed Ubuntu. There is nothing I cannot do in my fields of interest; video editing, audio

editing, software development, etc. The only piece of software I was happy to pay for was Turboprint, which allowed me to use the full potential of my printer.

I'm now looking forward to the fibre optic network in Aussie – which will lift my paltry connection from 1.5mb/s to 90 mb/s. Then I can download every Linux flavour and do a comprehensive review and comparison. You can publish the results in about issue #283.

Michael Davies

DEFAULT SCANNER

Is there a way to change the default scanner. Currently using 12.04 and Sane Simple scanner. Since I have multiple printer scanners on my system, is there any way to change the default scanner? I have tried Document > Preferences, and can manually select the scanner; however I would like to chose another 'default' scanner. Anyone?

Silverfox68

WINDOWS USERS

Like many people, I started my computer life with a Sinclair Spectrum ZX48, then progressed onto Windows computers.

Like many windows users I am fed up with the way Microsoft treats its users, and I have switched to Ubuntu 12.04 LTS.

As a newbie, some of the terminology is baffling to say the least; how about an article or two for us newbies on the basics of Ubuntu.

Many Thanks for the articles I have read – which do help

Philip Ford

Ronnie says: *I'd say have a look through FCM#54-68 where we did a series of articles called 'Closing Windows' which showed the various *buntu equivalents for Windows things.*

Join us on:



goo.gl/FRTMI



facebook.com/fullcircle magazine



twitter.com/#!/fullcirclemag



linkedin.com/company/full-circle-magazine



ubuntuforums.org/forum display.php?f=270

FULL CIRCLE NEEDS You!



Without reader input **Full Circle** would be an empty PDF file (which I don't think many people would find particularly interesting). We are always looking for articles, reviews, anything! Even small things like letters and desktop screens help fill the magazine.

See the article **Writing for Full Circle** in this issue to read our basic guidelines.

Have a look at the last page of any issue to get the details of where to send your contributions.

Tuxidermy

MAJOR BRETT, WE'VE LOST COMMUNICATION WITH YOU FOR LONG. WHAT HAPPENED? WHAT ARE THESE IMAGES YOU'VE SENT? THEY'RE SO BLURRY. I'M SORRY, SIR. OUR SYSTEMS WENT DOWN A WEEK AGO AND NOTHING WORKED. IT ALL BEGAN WITH THE PLANET AND THE ALIEN ASTRONAUT. THE WHAT?!

WE THOUGHT IT WAS A DEAD ALIEN, BUT IT TURNED OUT IT WAS A STATUE OF ANOTHER SPECIES. MAYBE A SUPERIOR BEING.

I SEE.

BUT THERE WAS MUCH MORE TO SEE. SIR.



WE'VE FOUND A ROOM FILLED WITH EGGS. ALIEN EGGS, AND THE CREW STARTED TO FEEL INTRIGUED.



DEAR LORD! DID YOU RUN? DID YOU DESTROY THEM? TELL ME YOU'VE BURNED THE DEVILISH THINGS TO ASHES!

ACTUALLY, NO. WE WERE VERY CURIOUS ABOUT WHAT THEY WERE AND WHAT THEY COULD DO. SO WE WAITED FOR A WHILE AND ONE OF THE EGGS STARTED TO HATCH.

CODE RED! CODE RED! ARE YOU INFECTED? DID THE FACE HUGGER GET YOU? I'M CALLING A TASK FORCE RIGHT NOW!

NO NEED FOR THAT, SIR. IT WAS A HUGGER, THAT'S FOR SURE, BUT NOT THE KIND YOU THINK.



AND WHAT HAPPENED, THEN?

WELL, SIR, FOR STARTERS, OUR SYSTEMS GOT WAY BETTER.





Q&A

Compiled by Gord Campbell

If you have a Linux question, email it to: misc@fullcirclemagazine.org, and Gord will answer them in a future issue. Please include as much information as you can about your query.

Q Is it possible to install Ubuntu over Windows 8 instead of dual boot?

A Yes! It depends on the specific computer to some extent. It will probably have Secure Boot enabled, and you might want to go into the BIOS to turn it off. Ubuntu, Kubuntu or Xubuntu 14.04 should install nicely from there.

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

Q I support a bank of computers made available for free public access in a local bookstore; three of them are running Ubuntu 12.04 LTS, the fourth is a Mac running OS X 10.6. All are set to log in as Guest with limited access rights, useful because each restores itself to the default setup with each reboot or log-in.

I've been able to modify the Guest setup on the Mac, adding customized wallpaper, modifying the applications on the Dock, the

browser home page, etc. It required making my desired changes and then running a set of Terminal commands.

I haven't, however, been able to find instructions for doing the same sort of thing on the Ubuntu systems. Any suggestions where I should look?

A I have never used the guest account, so I'm not familiar with how much of its setup is inherited from a more powerful (and permanent) account.

You might be better off to look at a Linux Distro which is intended for kiosk usage, such as Porteus, Webconverger or Instant Webkiosk.

Q My laptop has hybrid graphics, but I'm struggling to get it working in Ubuntu 12.04.

A You will find the support is much improved in 14.04.

Q Upgraded to 14.04 and can't get xscreensaver to work on 2 of 3 computers. Works perfectly on one. Doesn't load on a clean install or on the 13.10 upgrade.

A (Thanks to *blroth* in the Ubuntu Forums) Edit the startup programs. Change "xscreensaver -nosplash" to "xscreensaver -no-splash".

Q Why doesn't Software Updater notify me of all the available updates?

A (Thanks to *bapoumba* in the Ubuntu Forums) Basically, updates are marked to be offered to only a percentage of users. Packages are checked against regressions or bugs, then are gradually offered to all users if no problem is encountered. This works with the Software Updater. apt-get does not check for phased updates, so it retrieves anything that can be upgraded from the repos.

Q I just did a clean install of Ubuntu 14.04 on an old Acer Travelmate 2410 laptop to replace Windows XP. Everything is fine except that it won't shut down, it just hangs on the shutdown screen.

A (Thanks to *cyberwizzard2* in the Ubuntu Forums) This issue lies in the wistron_bttns module which crashes and is supposed to manage the WiFi kill switch and "special" buttons on the laptop. Disable it by using the command:

```
sudo gedit
/etc/modprobe.d/blacklist.conf
```

Add this line at the end:

```
blacklist wistron_bttns
```

Save the file and reboot using Alt-SysRq reissub (type slowly!)

If that doesn't work, you may be able to find the module causing the problem in dmesg.



TOP NEW QUESTIONS AT ASKUBUNTU

* How to patch the Heartbleed bug (CVE-2014-0160) in OpenSSL?
<http://goo.gl/K53nFn>

* What is the meaning of 'i686' in Ubuntu?
<http://goo.gl/CJ7gA4>

* Is Wine what I think it is?
<http://goo.gl/s7kMjB>

* Am I using beta or stable?
<http://goo.gl/1VMXwk>

* Packages are removed when I use "apt-get install" with a hyphen after the package name
<http://goo.gl/s4FiPe>

* Do I need to enable TRIM if I have a non-Intel/Samsung SSD
<http://goo.gl/fTIf92>

* Why was the virtualbox package removed from the 14.04 repository?
<http://goo.gl/QVRUqq>

* Using cd command to navigate directories in Ubuntu which have special characters

<http://goo.gl/LwvvlC>

* Is there any way to see how much space is used for Ubuntu?
<http://goo.gl/ktti8>

* Is there any GUI app to create .desktop files?
<http://goo.gl/TtFtwD>

TIPS AND TECHNIQUES



TECHIES?

I think about 2% of the population feels confident that they can install an operating system, including thinking about partitions: they are the techies. If my guess is close, then almost 100% of the world's techies are already running Ubuntu or Mint.

In that case, the only way our favorite operating systems can gain market share, is to have "out of the box" systems sold with our favorite OS. It's happening, but it's terribly slow.

Your thoughts?

TAPE BACKUP

The company where I am consulting is planning to install a new server. We haven't made any decisions yet, but some flavour of Linux is a possibility. The current server has a tape drive, used for a weekly full backup and incremental daily backups. I haven't spent a lot of time searching, but I have not found a package for Linux which can use the same approach.

But maybe there is a better approach.

The company does not re-use tapes, which cost about \$80 per week. A tape drive for the new server would cost around \$2,000.

Instead, we could buy a Network Attached Storage (NAS) which allows switching the hard drive, as easily as changing the tape in the tape drive. A 2 TB drive costs about \$100. Without having exact numbers, I think we could do a full backup and a month of incremental backups on a 2 TB drive. And with gigabit Ethernet, the backup would run a lot faster than the tape drive. There are several open-source packages which would work nicely,

and they are available for both Linux and Windows Server.

Even better, we would be using industry-standard technology. If we use a tape drive, and it dies four years from now, there is no guarantee that we could buy a new drive which could read the backup tapes. Bye bye tape!



Gord had a long career in the computer industry, then retired for several years. More recently, he somehow found himself "The IT Guy" at a 15-person accounting firm in downtown Toronto.



Security

Written by Michael Boelen

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

From **Brian Hall**: Even after trying to search for information on Clamav, I can't really find any comparison of its ability compared to other proprietary Linux AVs. I guess the key factor in any antivirus is the, hopefully regularly updated, database behind the program, but is the Clamav database less comprehensive (because it's open source) than those used by the free versions of proprietary AVs?

MB: The question is interesting in several ways: We humans like to compare products. After that we try to make a decision on what choice is the best one for us. In the case of malware detections tools, obviously the detection rate would be something to use for the selection. However, we could also use the percentage of false positives (incorrectly marked as being malicious) or consider the price to be the most important factor. In other words, what makes a malware detection to be "well enough" or outstanding..?

In case of anti-virus tools, it's

important to consider the threats you are trying to protect against. If the tool is used for scanning a mail server, each intercepted e-mail with malware is a win. Ones that are not picked up, well, end up in the user's mailbox, and hopefully don't get opened, or are properly detected by a local on-access virus scanner. In the case of an on-access scanner for surfing the web, you'd rather have a much higher detection rate.

Back to ClamAV.. ClamAV uses a core database, with a daily addition to it. This smaller daily database (daily.cvd) is regularly updated during the day. However, that doesn't say much about the detection rate. This is where professional comparison tests come into play. Unfortunately ClamAV often is not included in tests, because it's not commercial or not focused mainly on Windows.

However, there is no need to think ClamAV is not good due to lack of evidence. Because the project is community driven, and many people provide samples they

discover, it shouldn't take long for ClamAV to protect against new threats. Sometimes this occurs because another malware tool (correctly) discovered a new threat. Other vendors, including ClamAV, then include a signature to their database as well.

One of the best examples for "community driven malware detection", is the website VirusTotal. All submitted examples get analyzed, and results get shared with all participating vendors. So, if you discover a malware sample, and upload it, it may get recognized by only a few in the beginning, while after a few hours many of them "suddenly" recognize it. ClamAV is participating in this list of vendors, so it should benefit from submitted samples as well.

Even if the ClamAV database is less comprehensive than from other vendors, it depends on your use of the tool. With information security, we should never rely on just a single defense, but build a fortress of layers. Using a

community driven tool is just one of the possible layers we could add. From my personal experience, I can tell it helped many of my customers and their mailboxes. I'm sure it didn't detect every threat, but no single other software tool would be able to do that either.

If you have a security-related question, email it to: misc@fullcirclemagazine.org, and Michael will answer them in a future issue.

Please include as much information as you can about your query.



Michael Boelen is the author and project lead of *Lynis*. His company *CISOfy* provides security guidance to individuals and companies by sharing open source software, support and knowledge. He loves sport, reading, and enjoying life with friends.





Open Source Design

Written by Jens Reuterberg

One of the classical issues with Open Source as a social movement is its opinion on promo work and marketing. We distrust marketing. We see it as someone trying to trick someone else into using something that isn't very good.

In response, we often try to just describe what it is we have – assuming that its technical excellence will, in and of its own, make people use it.

We employ a communicative form in marketing and promo material that is perhaps “precise” but not very “correct” – a distinction that can be illustrated by, for example, calling this column “a longer bit of text defined as a column with more or less subjective commentary on the state of Open Source in general and design work in KDE community in specific”. It doesn't roll off the tongue does it? In fact, I got bored writing it halfway through, meaning that most would be bored reading it after the second word. A more correct term for this column

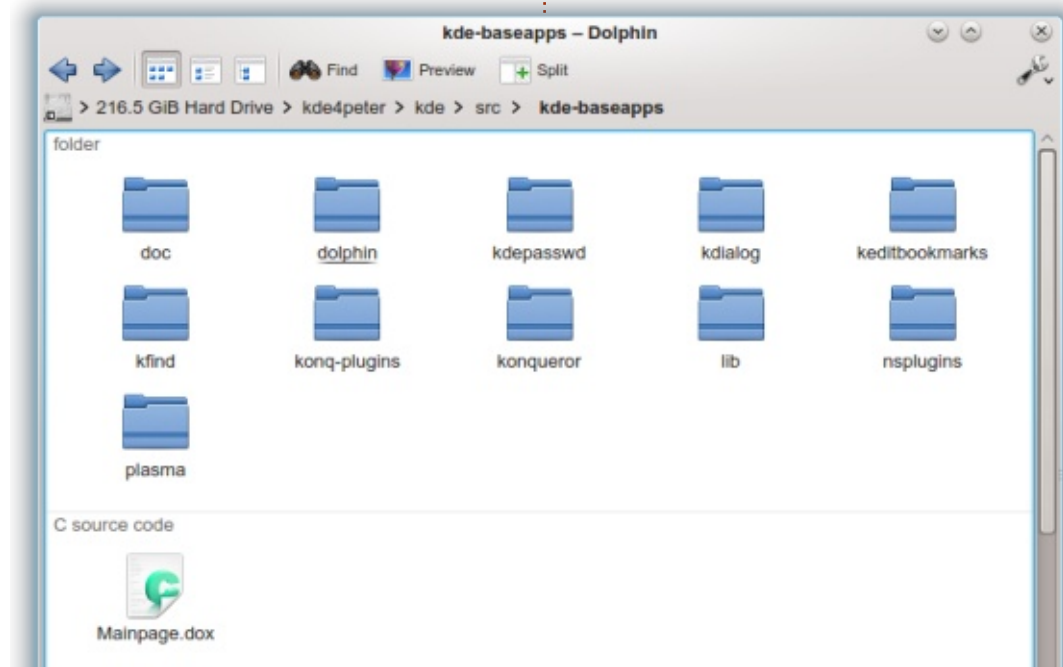
would be “Column about Design and Open Source” – it says what it needs to say but also without going into frivolous specifics.

So in promotional work, Open Source projects strive for exactness and being precise – we try to explain exactly what something is in the hope that the reader won't fall asleep about halfway through, and will notice the technical excellence that we have created. The bitter reality of which is that it doesn't work.

In communicative theory, it would be explained as based on the false presumption that the readers/receivers are all a homogeneous mass and that they share the writer's/sender's disposition exactly. So what can we do to change this without, for that matter, scrapping our healthy skepticism of “marketing language” (as someone who's had long experience with marketing – skepticism is a must)?

The first bit is throw everything we think we know about terminology out the window. A name, whether its an app, an OS, or the header for an article, has to be correct instead of precise. It has to deliver usable information based on assumptions of the reader. Saying “Filemanager” is better than calling it “Dolphin” even though “Dolphin” is the name of the application. Calling it “Files” is even better. Yes, the application is a filemanager, but who cares? Will a user, well versed in what Dolphin does, be confused by the application's new name? Probably not. The icon is there to assist you, the name is there telling you essentially what it is – so he or she won't even notice after the first two second confusion. But for the new user this makes sense. Files is where the files are, that's where they are handled - “managed” even. The assumption is there, so no need to spell it out and no need to be “precise” – just “correct”.

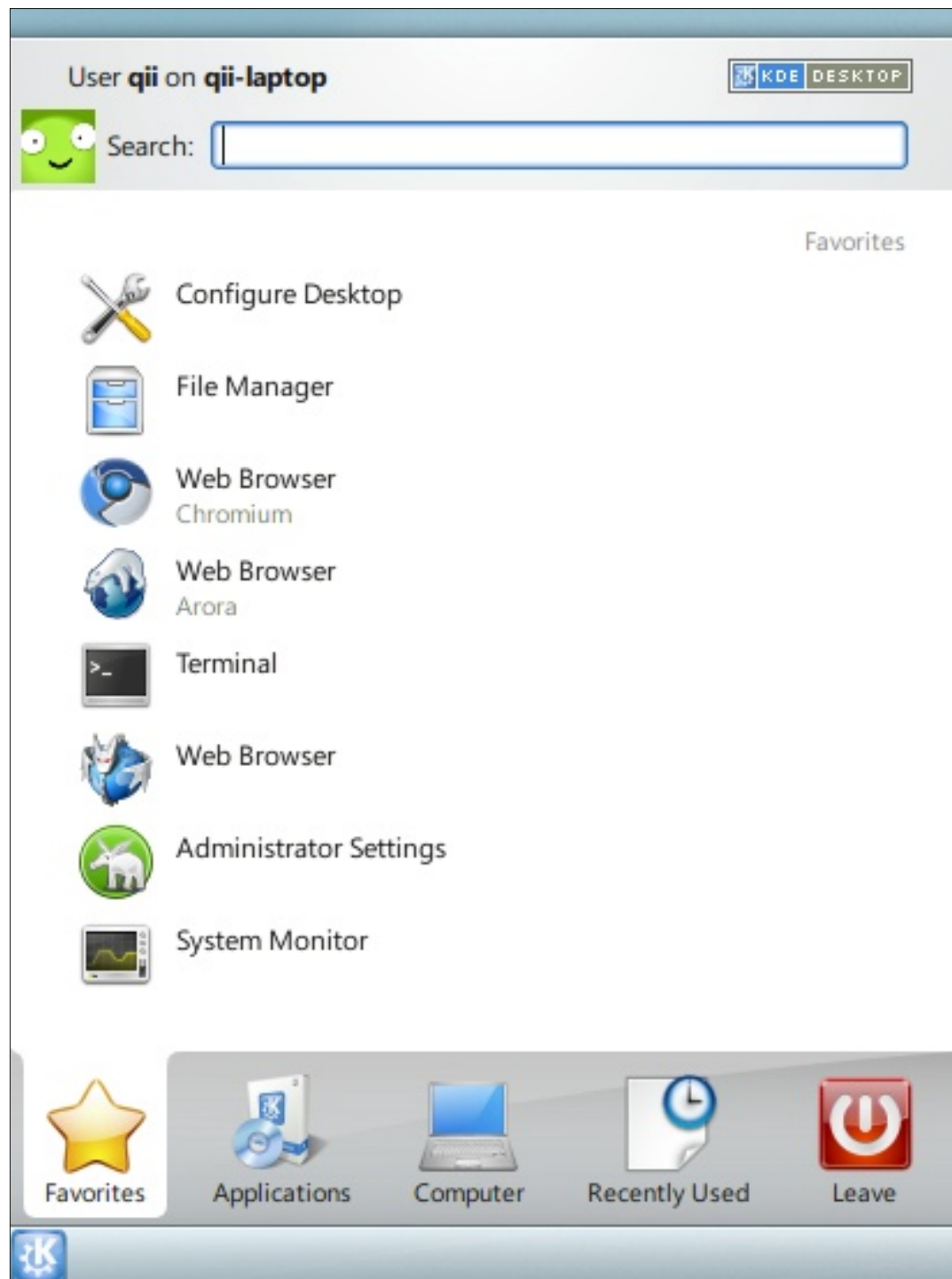
Another example of this is if we look at the actual marketing and promo being done. How many



articles haven't most linux users read that include more technical spec than information? Yes, technical specs ARE information, but it's information that requires a lot more from the reader than would be advisable. This is a good moment to step back and look at what is meant to be communicated in the text – first you want to add the soft core values of the thing being explained. After that you explain how these values are represented as actual details of the object, and, finally, after all that is done, you put in technical details.

Why do it that way, you might ask? Well, because, to most users, the most relevant bits are the core values of the desktop: "It's lightweight"; after that they might want to know WHY it's lightweight: "Because we removed applications that started by default before," and finally a technical summary of the changes done for those interested. A technically interested reader might skip ahead as long as he or she knows the technical bits are further down. But a technically disinterested person tends not to bother skimming through often more bulky technical bits to find the, to him or her, relevant parts.

But why should we even care about marketing? Simply put, user spread. We are writing texts for no one, designing applications for experts, and then seem to be shocked that many users feel alienated. By making it accessible to more, by leaning towards the assumption that our techy users feel covered enough, we can actually try to explain the awesomeness of the techy bit for those users who aren't at home in these surroundings. By doing that, we aren't dumbing things down – we're smarting it up. We are letting more people get an "in" to the technical brilliance that is Linux.





The Impossible Game is a newly released indie game developed by Grip Games and published by FlukeDude. The Impossible Game was released on May 8, 2014 through Steam for Microsoft Windows, Mac OSX and Linux. Originally, the game was released for the Xbox Live Arcade, but it's now available for almost all video game platforms. As the title suggests, it is a game that seems pretty impossible to beat. However, this game is pretty hard to put down. The Impossible Game is a co-ordination platform game with only five very difficult levels. Also included is a level editor which lets you create your own levels.

GETTING AND PLAYING THE IMPOSSIBLE GAME

The Impossible Game is available on Steam for a regular price of \$4.99, and it's also available through the Humble Bundle Store for about the same price. The Humble Bundle Store gives you the option to redeem it in the form of a Steam key which is

what I did and what I recommend. At the time of this writing, I actually paid \$2.99 for The Impossible Game from the Humble Bundle Store. If you haven't checked out the Humble Bundle Store, I strongly recommend doing so because they have a very large selection of games for Linux, and, best of all, they're usually on sale for a much lower price than from other outlets.

The Impossible Game is very easy to understand, yet very hard to master. Your one and only control is the jump button. With the space bar (or a left mouse click), you control an orange square

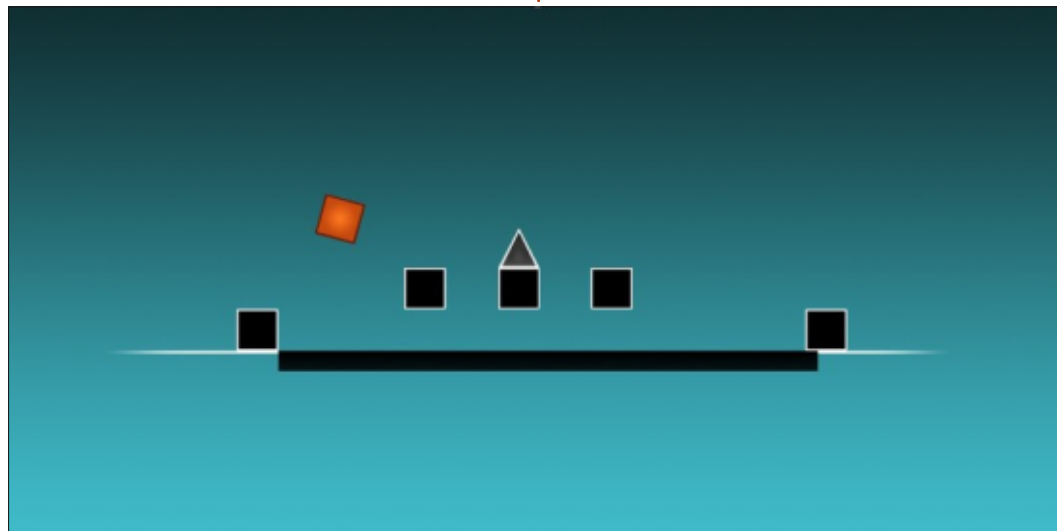
that has to avoid obstacles as it moves from left to right until it reaches the end of the level. Tapping the space bar makes your orange square jump over spikes or onto square blocks – both of which have the potential to destroy your square if you don't avoid them at the right time. Holding the jump button makes your orange square jump for longer if needed. It's all about timing your jumps just right. You can also place flags in the game as sort of check-points. Automatically, when you drop the first flag, your game switches to “practice mode” and becomes much easier to advance. In fact, this is the only way I've been able to

complete any levels, in practice mode.

The soundtrack is another key element of the game. As you advance in the game, the soundtrack is synced with your movements – which makes the game even more enjoyable. If your square happens to be destroyed, the music is also killed and unfortunately you have to re-spawn from the beginning of the level which can be very frustrating, especially if you were almost at the end of the song – which is also the end of the level. All in all, there's nothing too fancy, it's pretty simple but also very effective.

If the game's five levels are too “impossible” for you to beat, then you can create your own level. The Impossible Game also has a built in editor that lets you create your own levels any way you want. You can even use your own music for your own custom levels.

MY GAMING SETUP & MINIMUM REQUIREMENTS



I played The Impossible Game with my custom made desktop PC consisting of an AMD FX-6100 3.3GHz CPU, an Asus M5A97-EVO motherboard, a Sapphire Radeon HD 5770 graphics card, 8GB of Kingston Hyper X RAM, and a 1TB Seagate Barracuda hard drive. The software used was Ubuntu 14.04 LTS with Unity desktop and AMD 13.35 proprietary graphic drivers.

The minimum system requirements for The Impossible Game on Linux are:
OS: Ubuntu 12.04
Processor: 2 GHz
Memory: 2 GB RAM
Graphics: Open GL 2.0 Support
Hard Drive: 150 MB available space

CONCLUSION

Although the game is actually called The Impossible Game, you still wouldn't expect it to be as "impossible" as it is. I honestly do feel that it might be impossible to beat which can be pretty frustrating for someone like me who likes a feeling of accomplishment after working on something for a long time. I encountered some very minor glitches, but overall it was a very

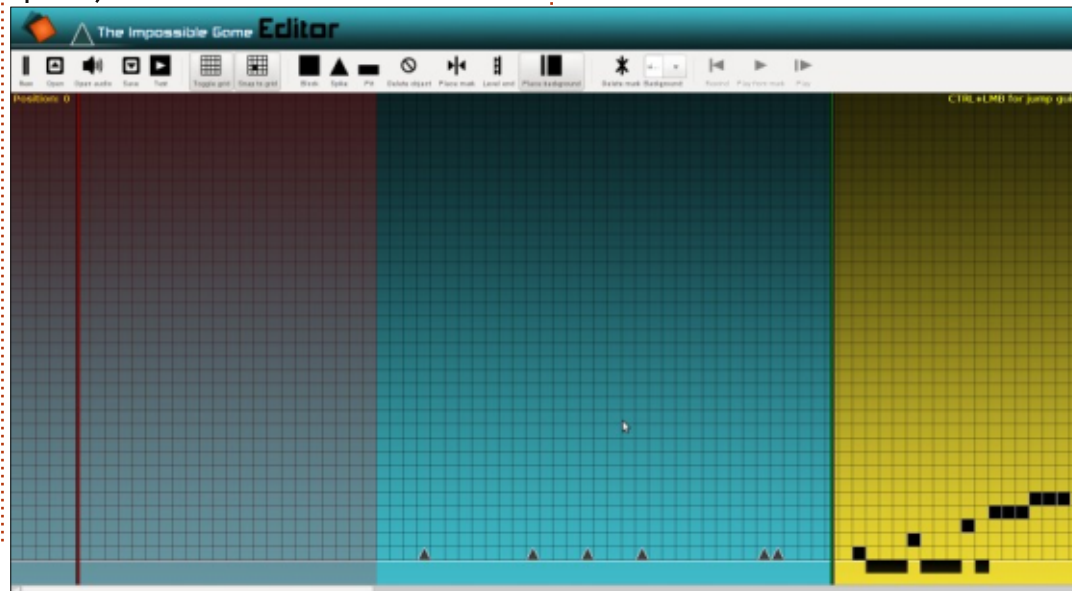
rewarding experience. It's a great game and I definitely recommend it, but I cannot give it a perfect score.

Pro's

- Easy to understand and easy to play (only one button)
- Challenging and entertaining, makes you come back for more
- Only five levels (usually this would be a con, but not for a game in which it's impossible to complete even one level)
- Level editor for you to create your own levels
- Soundtrack is an integral part of the game
- Easy to install
- Low price (only \$4.99 regular price)

Con's

- I think it may actually be impossible to beat (I have been able to complete only the first couple of levels in practice mode)
- A glitch in the game awarded me a medal for playing over 500 consecutive attempts after only 108 NON-CONSECUTIVE attempts
- Graphics are not that appealing, in fact they're pretty basic; maybe some interesting backgrounds would have been nice



COMPETITION

I've got Humble Bundle gifts to the first three readers who can email me the answer to this question:

what do you press to jump in The Impossible Game?

Email your answers to:
7bluehand@gmail.com



Oscar graduated from CSUN, is a Music Director/Teacher, beta tester, Wikipedia editor, and Ubuntu Forums contributor. You can contact him via: www.gplus.to/7bluehand or email: www.7bluehand@gmail.com



Strike Suit Zero isn't a game for the faint-hearted; it gets tough really quickly. Being a challenge and a compelling space shooter, it's also the same thing that will push away players as well.

The game takes patience, the levels will make you try and try again. If this isn't your thing, you might want to miss this one out. But if you are willing to push through and have the challenge of the chaos ramped up to the most punishing difficulty, the game is definitely rewarding.

You are in control of the Strike suit. While in your first few encounters you will be in a little interceptor jet, the Strike Suite adds an extra dynamic to the game. You are almost always outnumbered but never outgunned. With enemy fighters that can't stop the Strike Suits fury, it creates a sense of empowerment unlike any other space combat game. Strike Suit Zero has nailed the balance between simulation and arcade shooters; this helps to overlook the flaws that it does

have.

The story feels like it has already been told a hundred times. You play as a pilot named Adams, While on a routine mission to prove you're ready to get your wings again, the evil colonial forces attack the Earth fleet, destroying all but a few capital ships and star fighters. You gain your wings back and are given missions.

Despite creating engaging battles, Strike Suit Zero could have excelled with more missions and enemy variety. Every stage boils down to either defending a friendly

ship, attacking an enemy ship, or both together. Other games have just as formulaic an approach to mission design, but this is masked with better storyline, encounters and level design. Within the first few missions, you've basically seen all the game has to offer. The combat does remain fun throughout, but it would be great to get more varied missions and enemies to fight.

There are, on occasions, brutal checkpoints that ensure you'll replay large portions of Strike Suit Zero's missions. Also you can't save mid-mission, and with some of

them taking more than 30 minutes to complete, it does sometimes feel like a lot to ask when there are constant waves of enemies; it's sometimes off-putting to the not so hardcore players. There are plenty of incentives to play levels again after completing the campaign. On top of awarding additional weapons based on your score, each stage has a secondary objective to complete to unlock permanent upgrades for your ships. They are good and so is the combat, which is also worth popping back to older stages once you have some more skills.

In conclusion, Strike Suit Zero has nailed the arcade shooting and simulation, but is knocked back with repetitive missions and a lackluster story that makes you not care about the characters within the game. With some patience, the game can be very rewarding and you will gain some really good skills to tackle the harder levels.





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.



Right now I am using Mint KDE 16-64 on a homemade computer with: 8-core AMD FX-8350 CPU running at 4.4GHz instead of 4GHz, Asus M5A97 R2.0 motherboard, 16GB DDR3 RAM, Asus GTX760 Videocard with 2GB memory, 2 1TB disks plus (from an older computer) 2 500GB disks.

Jan Mussche

I am sending you my desktop screen as it is now. It consists of the following:
Window decorations: OSX_Aurorae
Cursor theme: Oxygen Blue

Desktop Theme: Tragedy
Desktop picture: Monument Valley, Arizona
Desktop widgets: 3 analog clocks, yaWP Weather Forecast widget

and the Shutdown button
Full HD 1920 x 1080
Organized the icons to disturb the picture as little as possible



I'm running Ubuntu 14.04; I added Numix theme and Numix-circle icons from Numix ppa, on a

Thinkpad T43p, Pentium M 2.13 GHz, 2GB Ram, 80GB hard disk, GPU Ati/Amd Fire GL V 3200.

Gabriele Tettamanzi



HOW TO CONTRIBUTE

FULL CIRCLE NEEDS YOU!

A magazine isn't a magazine without articles and Full Circle is no exception. We need your opinions, desktops, stories, how-to's, reviews, and anything else you want to tell your fellow *buntu users. Send your articles to: articles@fullcirclemagazine.org

We are always looking for new articles to include in Full Circle. For help and advice please see the **Official Full Circle Style Guide**: <http://url.fullcirclemagazine.org/75d471>

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: fullcirclemagazine.org

FCM#86

Deadline:

Sunday 08th June. 2014.

Release:

Friday 27th June. 2014.



Full Circle Team



Editor - Ronnie Tucker

ronnie@fullcirclemagazine.org

Webmaster - Rob Kerfia

admin@fullcirclemagazine.org

Podcast - Les Pounder & Co.

podcast@fullcirclemagazine.org

Editing & Proofreading

Mike Kennedy, Gord Campbell, Robert Orsino, Josh Hertel, Bert Jerred, Jim Dyer and Emily Gonyer

Our thanks go to Canonical, the many translation teams around the world and **Thorsten Wilms** for the FCM logo.

Getting Full Circle Magazine:



EPUB Format - Recent editions of Full Circle have a link to the epub file on the downloads page. If you have any problems with the epub file, you can drop an email to: mobile@fullcirclemagazine.org



Issuu - You can read Full Circle online via Issuu: <http://issuu.com/fullcirclemagazine>. Please share and rate FCM as it helps to spread the word about FCM and Ubuntu Linux.

