Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

ISSUE #88 - August 2014
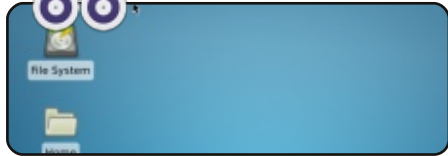
Photo: camknows (Flickr.com)

BLOATED UBUNTU DESKTOP?
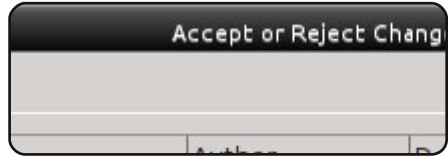## GET YOURSELF A SLEEK MINIMAL DESKTOP

Full Circle

THE INDEPENDENT MAGAZINE FOR THE UBUNTU LINUX COMMUNITY

HowTo

Graphics

```
#An alias to make the ls
command more detailed
alias ls = "ls -la --
color=always --classify"
```

BACK NEXT MONTH

BACK NEXT MONTH

EDITORIAL

WELCOME TO ANOTHER ISSUE OF FULL CIRCLE.

There's no Python this month, but there's an excellent replacement article on how to create a sleek minimalistic Ubuntu desktop. Great for speeding up your currently powerful PC, or for brining an old disused PC back to life. LibreOffice, Blender and Inkscape all continue, while the GRUB series comes to an end. But, where one thing ends another begins; a new series from Alan Ward on what a kernel is, does, and how you can compile one. Kernels are a complex beast, so expect this series to run for a good four or five issues at least. And if you're looking for a way to digitise your DVD collection to your media player then this month's column from Charles should keep you right.

I'm running low, or out of, several types of articles. So, if you'd like to see your words in FCM now is your best chance to submit; a review (hardware, software, books, etc.), questions about the security of your machine/software, a screenshot and some info about your current desktop, your personal story of how you become a Linux/Ubuntu user, even your opinion of Ubuntu, or the Linux world in general. As I always say; *if there's no articles, there's no magazine.* I like publishing your non-technical articles. It stops the magazine being too techy/geeky. Not that I have enough How-To's to fill the magazine each month, but I personally hate magazines that don't focus on the ordinary users. I want to know what software you folks are using, what games you're playing, what books you're reading, and what you're using your machine for and, if possible, how you did it.

All the best, and keep in touch!
Ronnie
ronnie@fullcirclemagazine.org

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

AUDIO MP3   AUDIO OGG

Download

Red Hat aims to standardize Linux for 64-bit ARM servers

Red Hat has developed a version of the Linux operating system that can be used to test chips and associated hardware based on the ARMv8-A 64-bit architecture for servers with the aim of standardizing that market.

http://goo.gl/RlBTqV [computerworld.com]

Submitted by: **Peter Odigie**

Does Oracle Linux 7 Give Larry A Cutting (Open) Edge?

Oracle has this month introduced the Oracle-flavoured Linux 7 open source operating system. Freely distributed under the GNU General Public License (GPLv2), Oracle Linux is based on Red Hat Enterprise Linux (RHEL) and follows the RHEL7 release, which arrived this June.

http://www.forbes.com/sites/adrianbridgwater/2014/08/04/does-oracle-linux-7-give-larry-a-cutting-open-edge/

Submitted by: **Peter Odigie**

Peppermint OS 5: Light, Refreshing Linux

The Peppermint OS is built around a concept that may be unique among desktop environments. It is a hybrid of traditional Linux desktop applications and cloud-based apps.

Using the Ice technology in the Peppermint OS is much like launching an app on an Android phone or tablet. For example, I can launch Google Docs, Gmail, Twitter, Yahoo Mail, YouTube, Pandora or Facebook as if they were self-contained apps on a mobile device -- but these pseudo apps never need updating. Ice easily creates a menu entry to launch any website or application as if it were installed.

This innovative approach puts the latest release of Peppermint OS 5, which appeared in late June, well ahead of the computing curve. It brings cloud apps to the Linux desktop with the ease and flexibility of a Chromebook. It marries that concept to the traditional idea of having installed software that runs without cloud interaction.

Source: http://www.linuxinsider.com/story/Peppermint-OS-5-Light-Refreshing-Linux-80859.html

Submitted by: **Jack M. Germain**

GUN Linux: On the range with TrackingPoint's new AR-15s

Since first running into TrackingPoint at CES 2013, we've kept tabs on the Austin-based company and its Linux-powered rifles, which it collectively calls "Precision Guided Firearms," or PGFs. We got to spend a few hours on the range with TrackingPoint's first round of near-production bolt-action weapons last March, when my photojournalist buddy Steven Michael nailed a target at 1,008 yards—about 0.91 kilometers—on his first try, in spite of never having fired a rifle before.

A lot of things have changed in the past year for TrackingPoint. The company relocated its headquarters from within Austin to the suburb community of Pflugerville, constructed an enormous manufacturing and testing lab to scale up PGF production, shed some 30 employees (including CEO Jason Schauble and VP Brett Boyd, the latter of whom oversaw our range visit in 2013), and underwent a $29 million Series D round of financing. It also sold as many PGFs as it could make, according to Oren Schauble, TrackingPoint's director of marketing and brother of former CEO Jason Schauble.

NEWS

Submitted by: **Lee Hutchinson**

Russian Ministry of Health to Replace Microsoft and Oracle Products with Linux and PostgreSQL

The Russian government is considering the replacement of Microsoft and Oracle products with Linux and open source counterparts, at least for the Ministry of Health.

Russia has been slapped with a large number of sanctions by the European Union and the United States, which means that they are going to respond. One of the ways they can do that is by stopping the authorities from buying Microsoft licenses or prolonging existing ones.

According to a report published on gov.cnews.ru, the official website of the Russian government, the Ministry of Health intends to abandon all the proprietary software provided by Oracle and Microsoft and replace it with open source software.

Submitted by: **Silviu Stahie**

elementary OS Freya Beta Is Out, Still the Most Beautiful OS in the World

elementary OS Freya Beta has been announced by its developers and it comes with an Ubuntu 14.04 base and lots of new features. As you can imagine, there are quite a few changes and improvements over elementary OS Luna, including the Linux kernel from Ubuntu 14.04, the 3.13 stack. This is just the tip of the iceberg.

elementary OS developers are supporting Facebook, Fastmail, Google+, Microsoft, and Yahoo account integration by default. This is done with the help of Pantheon Online Accounts, a new tool that combines features from Ubuntu Online Accounts and GNOME Online Accounts, and brings its own improvements.

This is still a Beta release, which means that users will probably notice bugs with the operating system. The release date remains unknown, but that is not something new. The developers never provide a release date and they usually take their time until they are satisfied with the result.

Submitted by: **Silviu Stahie**

Ubuntu Shopping Lens Declared Legal in the UK and Most Likely in the European Union

The UK authorities have declared that the Ubuntu Shopping Lens are legal and that no laws have been broken, either in Great Britain or in the European Union.

Canonical took a lot of flak from the community when the developers decided to integrate the Shopping Lens into the Ubuntu operating system. Two years have passed since then and a lot of things have changed in the meantime.

When the Ubuntu Shopping Lens were first introduced, users didn't have any control over them, at least not in a clear and easy way. There was no warning that data was sent over the network and there was no button to turn it off.

Currently, very few people even mention the Shopping Lens, and that is a clear sign that users have gotten used to them and that they have learned to use them or shut the functionality off entirely.

Submitted by: **Silviu Stahie**

WIRELESS SPEAKERS STREAM AUDIO FROM WEB AND WLAN

Denon debuted a line of Sonos-like wireless multi-room HiFi speakers that stream audio from both Internet and local sources, and run on embedded Linux.

Like the similarly Linux-powered devices available from Sonos, Denon's "Heos" wireless streaming speakers offer multi-room (multi-speaker) synchronized audio, and can deliver multiple audio streams from disparate sources to individual speakers or stereo-configured speaker pairs distributed around the home. Subscription streaming sources initially offered by Denon include Rhapsody, Pandora, Spotify, and TuneIn, with additional services offering DRM-free tracks "coming soon," says the company.

Source:
http://linuxgizmos.com/wireless-speakers-stream-audio-from-web-and-wlan/

Submitted by: **Arnfried Walbrecht**

THE SECURITY FLAWS IN TAILS LINUX ARE NOT ITS ONLY PROBLEM

If I ran a spy agency, the users of Tails Linux would be among the people I most wanted to spy on. Simply by using Tails, they have declared to the world that they want to hide something. As a spy, I would try to trick people into downloading a spyware-infested copy of Tails.

A great way to do that, would be to create a scam copy of tails.boum.org. An evil twin, if you will.

One of the most recent NSA revelations by Glenn Greenwald, on July 14th, was about GCHQ's Joint Threat Research Intelligence Group (JTRIG). Greenwald published a catalog of their assorted tools and techniques. One of the tools in the catalog is called HAVOK. It is the second item on page 8 of the document. HAVOK does "real time website cloning with on-the-fly alterations."

Source:
http://blogs.computerworld.com/netork-security/24179/security-flaws-tails-linux-are-not-its-only-problem

Submitted by: **Arnfried Walbrecht**

'MAYHEM' PROLIFERATES THROUGH WEB-SERVERS FREEBSD, LINUX, STATE EXPERTS

Security researchers have spotted one new malware that is named "Mayhem" as it proliferates through FreeBSD and Linux Web-servers, published theregister.co.uk dated July 18, 2014.

Security researchers Evgeny Sidorov, Konstantin Otrashkevich and Andrew Kovalev from Yandex, an Internet company in Russia, conducted a study of both the CnC (command-and-control) and client-side servers of Mayhem followed with writing a paper for Virus Bulletin.

According to them, a partial assessment of Mayhem that MalwareMustDie research team published during May 2014 reveals that the malware has various functions which are of one conventional bot for Windows; however, it's capable of operating even when rights to systems are restricted. Securityweek.com published this, July 18, 2014.

Source:
http://www.spamfighter.com/News-19100-Mayhem-Proliferates-through-Web-Servers-FreeBSD-Linux-State-Experts.htm

Submitted by: **Arnfried Walbrecht**

CAN ANDROID BE MADE TRULY FREE AND OPEN SOURCE?

Android has taken the world by storm, but many open source advocates view Google's mobile operating system with a dubious eye. Can Android ever be made to be a truly free and open source operating system? Or is it too tied to Google's products and services? Ars Technica took a stab at creating a FOSS version of Android.

According to Ars Technica: Android is a Google product—it's designed and built from the

ground up to integrate with Google services and be a cloud-powered OS. A lot of Android is open source, though, and there's nothing that says you have to use it the way that Google would prefer. With some work, it's possible to turn a modern Android smartphone into a Google-less, completely open device—so we wanted to try just that. Completely open Android is possible, but it feels like a constant uphill climb. It's harder and sometimes impossible to find open source solutions for many tasks. Even if you do find something, it will probably be uglier and less capable than the latest stuff out of Google HQ. But if you're willing to deal with a few headaches and slog through the sparse app selection, you'll have a better handle on your privacy and be able to brag that you have a (mostly) open source phone.

Source:
http://www.itworld.com/open-source/429102/can-android-be-made-truly-free-and-open-source

Submitted by: **Arnfried Walbrecht**

FROM CLOUDS TO CARS TO KITCHENS, LINUX MAKING AN IMPACT EVERYWHERE

There's no operating system more ubiquitous than Linux. It's everywhere. It's even running in devices and computers you may not suspect—our cars, our cell phones, even our refrigerators. Linux supports businesses and organizations everywhere, and because it underpins open-source innovation, it is the platform of choice for new applications. Companies such as IBM and their work with organizations like the OpenPOWER Foundation are creating such new innovations as Big Blue's new scale-out servers running Linux and putting them in places all around us. In fact, eWEEK recently ran a slide-show depicting how prevalent the operating system is in the supercomputing space. Linux has quickly become the operating system of choice in the high performance computing (HPC) market, growing from relative obscurity 15 years ago to powering 97 percent of the fastest computers in the world. But its appeal is found in more than cost or choice. This list, compiled with assistance from IBM, provides some examples of where Linux is making an impact.

Source:
http://www.eweek.com/enterprise-apps/slideshows/from-clouds-to-cars-to-kitchens-linux-making-an-impact-everywhere.html

Submitted by: **Arnfried Walbrecht**

LINUX WILL NOT BECOME A GAMING PLATFORM, IT ALREADY IS ONE

The true measure of any great gaming platform is not the number of games available. Nor is it the need to have the same games as other competing platforms (the Playstation 4 doesn't need Mario games to be considered successful). And it really isn't even about how many total games are sold, though that certainly helps.

No sir. It is far more nebulous and subjective than any of that.

The measure of a great gaming platform is if people want to use it to play games on… rather than another platform. At least on occasion.

For example: The SEGA Genesis. That beautiful console sold substantially less than the Super Nintendo. But it was still an excellent console that people enjoyed playing on. Thus, a success.

By that measure – and that is the only measure I can think of that makes any sense – Linux qualifies as a successful (even great) gaming platform.

Source:
http://www.networkworld.com/article/2459726/opensource-subnet/linux-will-not-become-a-gaming-platform-it-already-is-one.html

Submitted by: **Arnfried Walbrecht**

A YEAR WITHOUT WINDOWS AND A NEW LOVE OF LINUX

A little over a year ago, back in April 2013, I set myself a challenge. For so many years I, like countless millions, had used Microsoft Windows for my computing needs… and nothing else. You see, I had heard a lot about Linux and had dipped an

experimental toe into its watery depths, but I still relied heavily upon Windows. So my challenge was to switch to Linux as my main operating system, whilst blogging about my experiences. I figured I would be in a better position to talk about Linux, if I actually used it properly, instead of hovering on the periphery looking in as an observer.

You can read the blogs I wrote way back when this all started here:
https://www.alansitsolutions.com/blog/tag/working-without-windows/

What drew me to Linux in the first place? Well, for one thing, my inner geek yearned to 'have a play', but I also wanted to find out if it was true that Linux is effectively immune to computer viruses. I had also heard that Linux is much more secure than Windows, which means those pesky hackers have a harder time too. To top it off, Linux is generally free and so is a lot of the software!

Source:
http://www.midsussextimes.co.uk/news/a-year-without-windows-and-a-new-love-of-linux-1-6204763

Submitted by: **Arnfried Walbrecht**

THE CLOUD MIGHT BE THE KEY TO THE TRIUMPH OF DESKTOP LINUX

There have been many musings about why Linux doesn't have larger market share on the desktop. It's a topic that keeps coming up, over and over again, year after year. This time around, Tech Republic speculates on how the cloud could be the key to Linux triumphing on the desktop.

According to TechRepublic: The cloudbook could very well be the thing that vaults Linux into the hands of the average user, without having to stake its claim on Chrome OS or Android. And with the Linux cloudbook in the hands of users, the door for the Ubuntu Phone will have been opened and ready to walk through. Convergence made possible and easy.

The desktop, the cloudbook, the phone.

Source:
http://www.itworld.com/open-source/430006/cloud-might-be-key-triumph-desktop-linux

Submitted by: **Arnfried Walbrecht**

GESTURE-CONTROLLED HOME AUTOMATION HUB RUNS LINUX

Pre-orders are open for "Ninja Sphere," a $329, gesture-controlled, home-automation hub featuring Arduino hooks, ZigBee controls, and location tracking.

Sydney, Australia-based Ninja Blocks was one of the earlier entries in the Linux home automation game. The startup's open source Ninja Block hub launched on Kickstarter in 2012, and began shipping in a more advanced version last October. The $199 Ninja Block Kit integrated a BeagleBone Black SBC and an Arduino-compatible microcontroller, and offered remote access via smartphone apps and a cloud service. Using a 433MHz RF radio, it controlled vendor-supplied sensor inputs including motion detectors, contact closures, temperature and humidity sensors, and pushbuttons.

Source:
http://linuxgizmos.com/gesture-controlled-home-automation-hub-runs-linux/

Submitted by: **Arnfried Walbrecht**

RESEARCHERS DEMONSTRATE BADUSB HACK ON LINUX AND WINDOWS

German security researchers have demonstrated how they could take over Linux and Windows PCs by manipulating the firmware of USB drives. On Linux, the researchers require administrator rights which they acquired through a vulnerability in the screensaver. The hack was demonstrated on the the hackers conference Black Hat. Earlier the researchers already announced they found a critical security vulnerability in USB. By manipulating the firmware of USB drives (BadUSB hack), it was possible to infect other USB devices and, e.g., monitor network traffic, keystrokes, steal banking information, etc, etc.The trick doesn't only work on Windows PCs but also on computers running

Linux, the german security researchers demonstrated. The malware acquires administrator rights by displaying the screensaver and intercepting the password with a 'password stealer' when the user types. With administrator rights, the infected USB device gains access to other USB drives.

Source:
http://linuxgizmos.com/gesture-controlled-home-automation-hub-runs-linux/

Submitted by: **Arnfried Walbrecht**

COMPARING VIRTUAL MACHINES AND LINUX CONTAINERS PERFORMANCE

I BM Research Division has published a paper comparing the performance of container and virtual machine environments, using Docker and KVM, highlighting the cost of using Docker with NAT or AUFS, and questioning the practice of running containers inside of virtual machines.

The paper authors ran CPU, memory, network and I/O

benchmarks against native, container and virtualized execution, using KVM and Docker as virtualization and container technologies respectively. Benchmarks also include sample Redis and MySQL workloads; Redis exercises the networking stack, with small packets and a large number of clients, while MySQL stresses memory, network and the filesystem.

The results show that Docker equals or exceeds KVM performance in every case tested. For CPU and memory performance, KVM and Docker introduce a measurable but negligible overhead, although for I/O intensive applications both require tuning.

Source:
http://www.infoq.com/news/2014/08/vm-containers-performance

Submitted by: **Arnfried Walbrecht**

INCREDIBUILD TO LAUNCH LINUX & ANDROID ACCELERATION SOLUTIONS

I ncrediBuild, the leading solution provider of software

development and continuous delivery acceleration technology, announces the launch and beta program of its new suite of offering to natively support and accelerate Linux and Android based builds and application development processes.

The new offering has been designed under the same architecture and business logic of the IncrediBuild for Windows products, with a strong emphasis on all features and functionality developers so appreciate in IncrediBuild, such as ease of use, off-the-shelf deployment to Linux and Android based development groups with immediate acceleration benefits.

"What we can already disclose is that we've kept to our roots with regard to simplicity and enhanced value. We're very pleased with the results and can't wait to make it generally available," says Eyal Maor, IncrediBuild CEO. The system has already been tested and approved by leading software vendors. IncrediBuild will shortly launch a formal beta program, and the product will be released in Q4. (Contact IncrediBuild at linuxbeta@incrediBuild.com to get

more information and register for the beta program).

Source:
http://www.prnewswire.co.uk/news-releases/incredibuild-to-launch-linux--android-acceleration-solutions-271196211.html

Submitted by: **Arnfried Walbrecht**

10 YEARS LATER, MUNICH MAY DUMP LINUX FOR WINDOWS

A few years ago, Linux sites were buzzing that the city of Munich, Germany was going to kick Windows to the curb and roll out a Linux-based OS on all their government desktops. Now it looks like politicians are seriously considering switching back to Windows.

There are a few different forces at work here, according to reports from German news outlets. When the transition was announced years ago, supporters claimed it was going to save the city truckloads of cash. As recently as July, figures in the tens of millions were thrown around — but critics have repeatedly questioned whether

those savings are real or whether they've been trumped up to make the move look like it wasn't a massive failure.

Employees haven't been too thrilled with the change, and there have also been difficulties finding Linux-compatible software titles that meet Munich's needs. That's a little surprising to me as an IT administrator. You'd think those holes would've been plugged prior to the migration… Then again, management doesn't always listen when the IT guy speaks.

Source:
http://www.geek.com/microsoft/10-years-later-munich-may-dump-linux-for-windows-1602234/

Submitted by: **Arnfried Walbrecht**

SYSTEMD: HARBINGER OF THE LINUX APOCALYPSE

Now that Red Hat has released RHEL 7 with systemd in place of the erstwhile SysVinit, it appears that the end of the world is indeed approaching. A schism and war of egos is unfolding within the Linux community right now, and it is

drawing blood on both sides. Ultimately, no matter who "wins," Linux looks to lose this one.

The idea behind systemd was to replace the aged Init functionality and provide a sleek, common system initialization framework that could be standardized across multiple Linux distributions. systemd promised to speed up system boot times, better handle race conditions, and in general, improve upon an item that wasn't exactly broken, but wasn't as efficient as it could be.

As an example, you might be able to produce software that could compile and run on numerous Linux distributions, but if it had to start at boot time, you could be required to write several different Init-style boot scripts, one for each supported distribution. Clearly this is inelegant and could use improvement.

Source:
http://www.infoworld.com/d/data-center/systemd-harbinger-of-the-linux-apocalypse-248436

Submitted by: **Arnfried Walbrecht**

XEN HYPERVISOR TARGETS AUTOMOTIVE VIRTUALIZATION ROLE

The Xen Project Collaborative Project has launched an Embedded and Automotive initiative to expand its virtualization technology beyond the datacenter and cloud realms. Initially, the subproject will center on a collaboration with GlobalLogic on the company's Nautilus [PDF] in-vehicle infotainment (IVI) and telematics platform. An embedded version of the open source Xen Project type 1 hypervisor is integrated in Nautilus to enable sandboxed implementations of Android (for IVI) and either QNX or Linux for telematics and other back-end automotive services. The QNX/Android combination appears to be the main focus, however.

The idea of using multicore processors to house separate, sandboxed containers in automotive computers has been tossed around for a half decade or more, but is only now beginning to approach reality. For example, last October, Mentor Graphics announced a Mentor Embedded Hypervisor that can field a mix of

OSes including Linux, Android, and the company's Nucleus real-time operating system (RTOS). Bringing a hypervisor to automotive systems reduces the number of processors and complexity needed in the system while retaining a secure firewall between consumer-facing Android or Linux IVI software and back end Linux or RTOS telematics systems.

Source:
http://linuxgizmos.com/xen-hypervisor-targets-automotive-virtualization/

Submitted by: **Arnfried Walbrecht**

EMULATOR BRINGS X86 LINUX APPS TO ARM DEVICES

Eltechs announced a virtual machine that runs 32-bit x86 Linux applications on ARMv7 SBCs and mini-PCs, and is claimed to be 4.5 times faster than QEMU.

The open source QEMU emulator has long been the go-to app for providing virtual machines (VMs) that mimic target hardware during development or otherwise

run software in alien territory. Every now and then, someone comes up with software that claims to perform all or part of QEMU's feature-set more effectively. In this case, Eltechs has launched its Eltechs "ExaGear Desktop," a VM that implements a virtual x86 Linux container on ARMv7 computers and is claimed to be 4.5 times faster than QEMU. Despite its "desktop" naming, we can imagine many non-desktop possibilities fpr ExaGear in embedded and IoT applications.

Source:
http://linuxgizmos.com/emulator-brings-x86-linux-apps-to-arm-devices/

Submitted by: **Arnfried Walbrecht**

IBM TAPS GLOBAL NETWORK TO FUEL LINUX

At the LinuxCon North America conference today, IBM (NYSE: IBM) announced it is tapping into its global network of over 50 IBM Innovation Centers and IBM Client Centers to help IBM Business Partners, IT professionals, academics, and entrepreneurs develop and deliver new Big Data and cloud computing software applications for clients using Linux on IBM Power Systems servers.

Last year IBM committed $1 billion in new Linux and open source technologies for its Power Systems servers including the opening of five new Power Systems Linux Centers in Beijing, China, New York, New York, Austin, Texas, Montpelier, France and Tokyo, Japan. Today over 1500 ISV applications are available for Linux on Power, fueled in part by work performed at these centers.

Source:
http://goo.gl/AUP0jxh
[marketwatch.com]

Submitted by: **Arnfried Walbrecht**

A 16-BIT RPG WITH A MODERN TWIST

Standing out in a crowded sea of retro-style indie games is no easy task, but 16-bit RPG Elysian Shadows caught my attention recently for what it's promising to do differently.

Fashioning itself as a "next-gen 2D RPG," Elysian Shadows isn't shy about wearing its inspiration on its sleeve. Chrono Trigger and Secret of Mana both get shout outs in the developer video, and the aesthetics are right out of 1995. Those names are enough to get any 16-bit fan excited, but they're also names I hear referenced in almost every video about a new retro game.

Better 2D/3D graphics, dynamic lighting and sound, a real-time day/night system, and a world where everything is interactive (no "decorative doors") are a few things that could push this from just another nostalgia trip into something genuinely memorable.

Source: http://kotaku.com/a-16-bit-rpg-that-promises-a-modern-twist-1626228329

Submitted by: **Arnfried Walbrecht**

CHINA DEVELOPING ITS OWN OS

If it hasn't been made clear enough in recent months that China would love nothing more than to cut down on its reliance to American technology companies, its just-announced decision to create its own operating system should remedy that. At first, this OS will target the desktop, but eventually, it'll make its way to smartphones and other mobile devices.

At this point, we know very little about what China's OS will look like, or be like for that matter, but we do know that it's being designed to be a proper replacement for Microsoft's and Google's OSes. It seems very likely that China's OS would use Linux as a base, since there's little point in reinventing the wheel, and because of its open-source nature, the country would have complete control over the code. Further, Linux natively supports both x86 and ARM architectures, so that'd help take care of both the desktop and mobile aspect of the OS, and of course, Linux already supports a lot of software.

Source:
http://hothardware.com/News/China-Developing-Its-Own-OS-To-Take-On-Apple-Microsoft-and-Google/

Submitted by: **Arnfried Walbrecht**

Last month's article was devoted to configuring grunt and node.js to allow for compiling of SASS. However, very little time was spent actually discussing SASS. This article should hopefully remedy that, as SASS can be extremely useful for any web programmer.

WHAT IS SASS?

SASS stands for Syntactically Awesome Style-Sheets, and is a pre-processor for CSS. It essentially adds features to CSS (such as nesting, functions, mixins, imports, custom units, mathematics, inheritance, and variables).

NESTING

Imagine you have a two types of links on your website - a "default" style that should be blue and underlined (the standard values), but then you have links in your menu that you want to remove the decorations from, and adjust the :hover settings to darken the color of the button. In normal CSS, it

could look something like this:

```
ul.menu li a { text-
decoration: none; }

ul.menu li a:hover {
background-color:
#000000; }
```

In SASS, however, you can do the following:

```
$bg-color: #0000FF;

ul.menu { li { a{ text-
decoration: none; &:hover {
background-color: darken($bg-
color, 15%);}}}}
```

It looks a little less legible when written in a single line. However, format it nicely, and it looks something like that shown top right.

A few other things I'm doing in this example:
• Defining a variable (i.e. the company color, or a primary color in a template). This variable can be used anywhere in the SASS file. This way, any change to the color scheme requires only a single edit.
• Using a SASS function called darken, which calculates the value

```
$bg-color: #0000FF;
ul.menu {
    li {
        a{
            text-decoration: none;
            &:hover {
                background-color: darken($bg-color, 15%);
            }
        }
    }
}
```

for a darker version of the supplied color. SASS offers many functions, such as the ability to create color values based on hue, saturation and lightness (hsl).
• &:hover - the ampersand essentially places the following text directly behind the parent when nesting. For example ul { .menu{}} would result in ul .menu in normal CSS (in other words, find an element with the class menu after a ul element), and ul { &.menu{}} would result in ul.menu. The second one means "find a ul element with class menu". Ergo, this example means "find each tag in the menu unordered list, and darken the color when the user hovers over it".

MIXINS

Mixins are essentially macros, or functions, that do no calculation, but instead turn one line of SASS into multiple lines of CSS. A prime example is using CSS gradients. As this is a feature implemented differently by nearly every browser, it requires many lines of code to ensure the gradient appears in every browser that supports it. A SASS mixin I use for this purpose, looks like the example on the next page, at the top.

Now, this does look a bit like gibberish. However, the second background: line can safely be ignored if you don't want to use an SVG colored background in IE9. Everything else is commented to indicate which version of the

contents ^

COMMAND & CONQUER

```scss
@mixin gradient($color1, $color2) {
    background: $color1; /* Old browsers */
    /* IE9 SVG, needs conditional override of 'filter' to 'none' */
    background:
url(data:image/svg+xml;base64,PD94bWwgdmVyc2lvbj0iMS4wIiA/Pgo8c3ZnIHhtbG5zPSJodHRwOi8vd3d3LnczLm9yZy8yMDAwL3N2ZyIgd2lkdGg9IjEwMCUiIGhlaWdodD0iMTA
wJSIgdmlld0JveD0iMCAwIDEgMSIgcHJlc2VydmVBc3BlY3RSYXRpbz0ibm9uZSI+CiAgPGxpbmVhckdyYWRpZW50IGlkPSJncmFkLXVjZ2ctZ2VuZXJhdGVkIiBncmFkaWVudFVuaXRzPSJ1
c2VyU3BhY2VPblVzZSIgeDE9IjAlIiB5MT0iMCUiIHgyPSIxMDAlIiB5Mj0iMTAwJSI+CiAgICA8c3RvcCBvZmZzZXQ9IjAlIiBzdG9wLWNvbG9yPSIjMzY3ODY0IiBzdG9wLW9wYWNpdHk9I
jEiLz4KICAgIDxzdG9wIG9mZnNldD0iMTAwJSIgc3RvcC1jb2xvcj0iI2ExYjRiMiIgc3RvcC1vcGFjaXR5PSIxIi8+CiAgPC9saW5lYXJHcmFkaWVudD4KICA8cmVjdCB4PSIwIiB5PSIwIi
B3aWR0aD0iMSIgaGVpZ2h0PSIxIiBmaWxsPSJ1cmwoI2dyYWQtdWNnZy1nZW5lcmF0ZWQpIiAvPgo8L3N2Zz4=);
    background: -moz-linear-gradient(top, $color1 0%, $color2 100%); /* FF3.6+ */
    background: -webkit-gradient(linear, left top, right bottom, color-stop(0%,$color1), color-stop(100%,$color2)); /* Chrome,Safari4+ */
    background: -webkit-linear-gradient(top, $color1 0%,$color2 100%); /* Chrome10+,Safari5.1+ */
    background: -o-linear-gradient(top, $color1 0%,$color2 100%); /* Opera 11.10+ */
    background: -ms-linear-gradient(top, $color1 0%,$color2 100%); /* IE10+ */
    background: linear-gradient(to bottom, $color1 0%,$color2 100%); /* W3C */
    filter: progid:DXImageTransform.Microsoft.gradient( startColorstr='#{$color1}', endColorstr='#{$color2}',GradientType=1 ); /* IE6-8 fallback
on horizontal gradient */
}
```

browser it supports, and the colors supplied in the parameters of the mixin ensure that every gradient looks the same. This is the defined mixin - as such, you need to type it only once. I typically put variables, functions and mixins at the top of the SASS file, but it doesn't really matter.

To actually use the gradient mixin (to create a CSS gradient), you simply need to type: @include gradient($color1,$color2);

Example:
@include gradient(#367864,#537a7c);

Results in the CSS shown right.

```css
background-color: #367864;
   background: #367864;
   /* Old browsers */
   /* IE9 SVG, needs conditional override of 'filter' to 'none' */
   background:
url(data:image/svg+xml;base64,PD94bWwgdmVyc2lvbj0iMS4wIiA/Pgo8c3ZnIHhtbG5zPSJodHRwOi8vd3d3LnczLm9yZy8yMDAwL3N
2ZyIgd2lkdGg9IjEwMCUiIGhlaWdodD0iMTAwJSIgdmlld0JveD0iMCAwIDEgMSIgcHJlc2VydmVBc3BlY3RSYXRpbz0ibm9uZSI+CiAgPGxp
bmVhckdyYWRpZW50IGlkPSJncmFkLXVjZ2ctZ2VuZXJhdGVkIiBncmFkaWVudFVuaXRzPSJ1c2VyU3BhY2VPblVzZSIgeDE9IjAlIiB5MT0iM
CUiIHgyPSIxMDAlIiB5Mj0iMTAwJSI+CiAgICA8c3RvcCBvZmZzZXQ9IjAlIiBzdG9wLWNvbG9yPSIjMzY3ODY0IiBzdG9wLW9wYWNpdHk9Ij
EiLz4KICAgIDxzdG9wIG9mZnNldD0iMTAwJSIgc3RvcC1jb2xvcj0iI2ExYjRiMiIgc3RvcC1vcGFjaXR5PSIxIi8+CiAgPC9saW5lYXJHcmF
kaWVudD4KICA8cmVjdCB4PSIwIiB5PSIwIiB3aWR0aD0iMSIgaGVpZ2h0PSIxIiBmaWxsPSJ1cmwoI2dyYWQtdWNnZy1nZW5lcmF0ZWQpIiAv
Pgo8L3N2Zz4=);
   background: -moz-linear-gradient(top, #367864 0%, #537a7c 100%);
   /* FF3.6+ */
   background: -webkit-gradient(linear, left top, right bottom, color-stop(0%, #367864), color-stop(100%,
#537a7c));
   /* Chrome,Safari4+ */
   background: -webkit-linear-gradient(top, #367864 0%, #537a7c 100%);
   /* Chrome10+,Safari5.1+ */
   background: -o-linear-gradient(top, #367864 0%, #537a7c 100%);
   /* Opera 11.10+ */
   background: -ms-linear-gradient(top, #367864 0%, #537a7c 100%);
   /* IE10+ */
   background: linear-gradient(to bottom, #367864 0%, #537a7c 100%);
   /* W3C */
   filter: progid:DXImageTransform.Microsoft.gradient(startColorstr='#367864',
endColorstr='#537a7c',GradientType=1 );
   /* IE6-8 fallback on horizontal gradient */
```

COMMAND & CONQUER

FUNCTIONS

Functions work much the same way as mixins. Define your function with:

```
@function name($param1,
$param2) { @return $result; }
```

And to use it in your SASS:
name($param1, $param2);

MATHEMATICS

SASS also adds options to use mathematical operators (such as *, +, /, -). You must always define units to your numbers, or else use a function to ensure that SASS knows what the unit should be. Otherwise it will result in invalid CSS (i.e. width: 150, instead of width: 150px).

INHERITANCE

This allows you to extend CSS defined elsewhere in your file. If, for example, you have warning messages and each one is intended to have a different color font, but otherwise remain the same, you can define it like this:

```
.message {
  border: 1px solid #ccc;
  padding: 10px;
  color: #333;
}

.error {
  @extend .message;
  color: red;
}
```

IMPORTING

Lastly, SASS also allows you to create separate files, and import them into a main file. Something like this:

```
_mobile.scss – for screen
sizes smaller than 720px
_desktop.scss – for larger
screens
_mixins.scss – to contain all
your mixins
_variables.scss – to contain
your variables
app.scss – the main file,
that only contains imports
for each of the other files
listed.
```

If anyone is looking for a SASS-enabled framework with various useful features (i.e. a grid, accordion, carousel, etc.), I have used the Zurb Foundation framework a few times. Otherwise, if you prefer to use your own CSS code, it might be time to convert it to SASS, to make further edits a little less time-consuming. And using the C&C articles from Issues

84-87, you can create a git repository with a configured grunt system to house your new SASS framework.

If anyone has any issues, questions, corrections, or requests, you are always welcome to send me a quick email at lswest34+fcm@gmail.com.

FURTHER READING:

http://sass-lang.com/guide - The official SASS guide. It covers some examples, and also houses the manuals (under Documentation).

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.

Full Circle Podcast Episode 41, Trusted To Fail!!

Welcome to our new format show, there are several changes from the previous format, the most important being we are now recording together at the Blackpool Makerspace in the office. This Episode we Test Ubuntu 14.04, Review of Official Ubuntu Server Book.

Your hosts:
• Les Pounder
• Tony Hughes
• Oliver Clark

from the
Blackpool (UK) LUG
http://blackpool.lug.org.uk

So Trusty Tahr is out, and, once more, it's time to think about upgrading our setup, or not. 14.04 is a Long Term Support (LTS) version, so it seems worthwhile even for those less inclined to keep on the "bleeding edge" - if only to stick with it and have some peace for the next several iterations. But while downloading the many hundreds of Megabytes the recent CD images weigh, I always have a tendency to wonder if all these software packages that come as standard are really needed. The full installation of a recent Ubuntu distribution takes up a mere 3GB of disk space. OK, so this is really light compared not only to other operating systems, but even to Linux distributions such as Fedora. But would it be possible to reduce this even further, while maintaining a usable desktop interface?

This is not just a philosophical question, though it may strike a cord in the heart of people who (still) believe in the old Unix adage: "Small is beautiful!" Consider: a hard disk installation using as little

space as possible will boot up as fast, if not faster than a more loaded one. Just having fewer configuration files and fewer lines in them will give the system a speed advantage when initializing services. Geeky shenanigans like booting up over a network are facilitated, but also the average user's work-flows such as booting from slow media (USB drives, SD cards) - or merely using older hardware.

There was a recent editorial on OMG!Ubuntu! titled "Arch: Shangri-La for Ubuntu Power Users?" (http://www.omgubuntu.co.uk/2014/04/arch-shangri-la-ubuntu-power-users). This article by Sam Tran suggested that advanced Ubuntu users could benefit from at least trying out the Arch distribution. Needless to say, the proposal gave the commentators a field day, pointing out that Arch's aims as a distribution are just about in direct opposition to Ubuntu's. Arch gives deeper insight into the gory details of the operating system, while Ubuntu tries to shield the less technical folk from

complete exposure to them. Arch gives the system administrator complete power over which software is installed and how it is set up, while Ubuntu hides some of the details to give the newcomer fewer hairy choices to make from the outset. In other words, an Arch installation will probably end up quite a bit more streamlined, but Ubuntu will cause less confusion for the non-specialist.

So, using the Arch distribution to create a reduced-footprint desktop would certainly be not only a rational choice, but also one with which we would be able to go further up that road and really reduce disk space. But, from a personal point of view, I do think Ubuntu and its derivatives are better placed to help a "normal" user achieve a rational setup without too much hassle. I use the word rational as in functional (it must work), but also reasonably lightweight (it must weigh considerably less than a standard installation).

CHOOSING A DESKTOP AND AN INSTALLATION IMAGE

It is no secret that among the different desktop choices available to Ubuntu users, some need more disk space than others. It will also come as no surprise that the more complete, but also slower, desktop interfaces are also those that take up more space. So Unity, Gnome and KDE are perhaps not the best suited for this particular task.

Among the lighter desktop session managers, LXDE used by Lubuntu and XFCE used by Xubuntu are perhaps the two best-known. Choosing between them is largely a matter of personal opinion and taste, so if I will be using XFCE it is mainly because it is what I feel comfortable with both for speed and for its appearance. Your choice could very well be different, especially if more exotic desktops such as the Enlightenment (used by Bodhi Linux) or Razor-qt (a lightweight alternative based on the same Qt libraries as KDE) are thrown into

the mix. However, to install these desktops, you will need to set up PPA repositories, so this would perhaps best be left for another occasion.

To get a minimized installation, two different approaches are possible: either we start out from a complete installation, and then remove what we do not want, or alternatively we can start out from a minimal installation and add just what we wish to use. With the apt package infrastructure that gives us the power to install or remove packages at will, both options should in theory be equivalent. However, in actual practice it is more convenient to choose the second route. This is why I will be starting out not with a desktop distribution image, but with a

server image. This is a specialized distribution that starts out by installing a very basic system, on top of which we will add just the packages we need to set up the graphical desktop.

Since I will be using this setup on several rather elderly low-powered laptops later on, I downloaded the i386 (32-bit) version of the server distribution for Trusty Tahr (14.04). Though it is perhaps not immediately obvious from Ubuntu's web page - that insists on offering the 64-bit version - the 549MB image for the i386 architecture can be downloaded from the following address:
http://releases.ubuntu.com/14.04/ubuntu-14.04-server-i386.iso.

INSTALLING THE BASIC SYSTEM

Once we have the ISO image burned to a CD or installed on an USB thumb-drive, we can use it to boot up the target machine. The main difference from desktop Ubuntu images is that we have no Live CD option: we do not get to test the distribution directly, but must install it to hard drive first.

The text-based installer (it actually uses the ncurses library) will get us through exactly the same stages as the better-known GUI installer. Using the manual tool, I simply created a single ext4 partition that used up the hard drive completely, though if in doubt the automatic partitioner

should also make a good job of it. I am using a virtual machine and hard drive to prepare this, but, if using a physical computer, the usual caveats apply: make a backup of any data you want to conserve beforehand, because complete partitions (and the operating systems they contain) may get completely erased.

At one point, the installation program will propose a choice of additional software packages. Most users will not need these, so we will leave them unchecked. They can always be installed later on if needed.

Once the system is installed, we can reboot from the hard drive and login at the prompt. We will have access to only a text-based
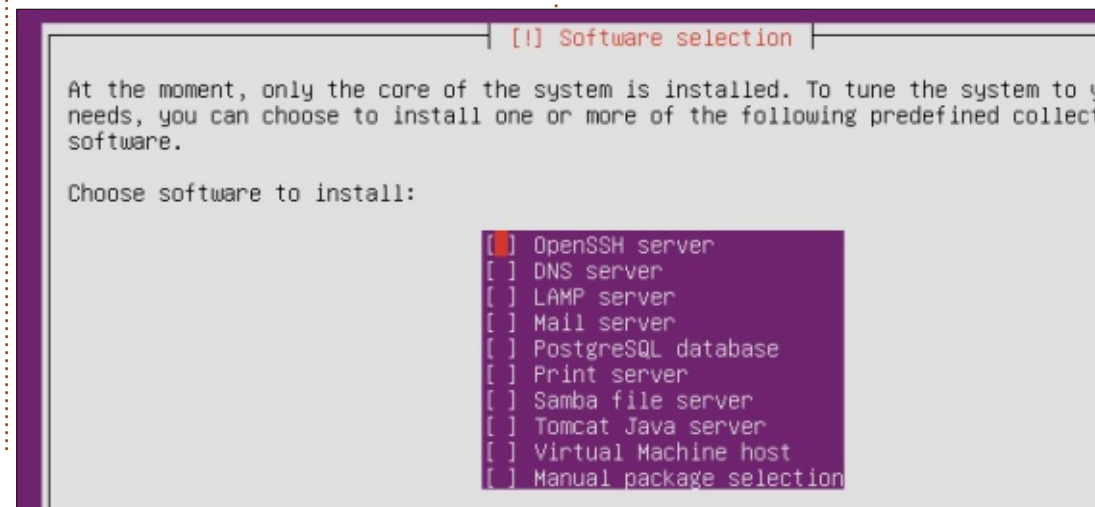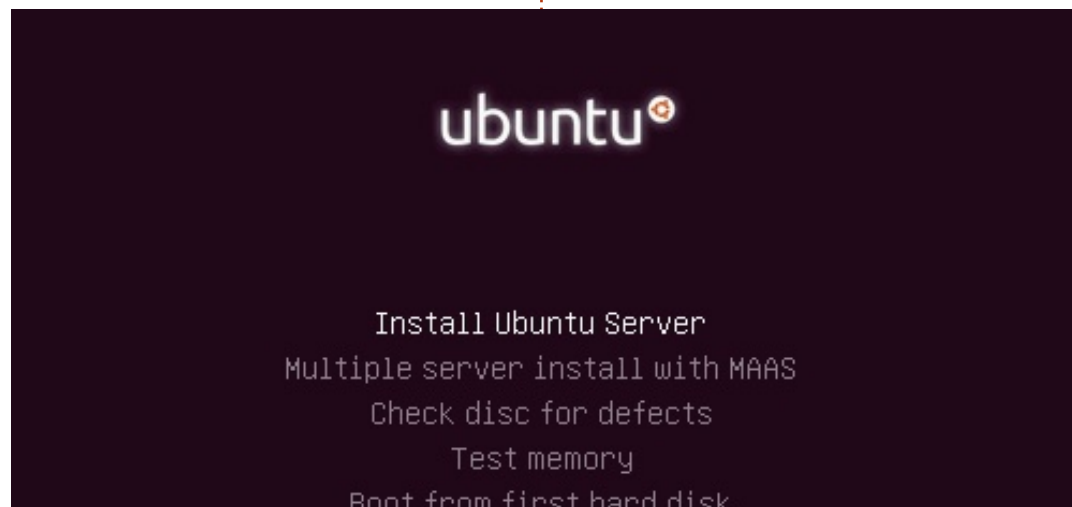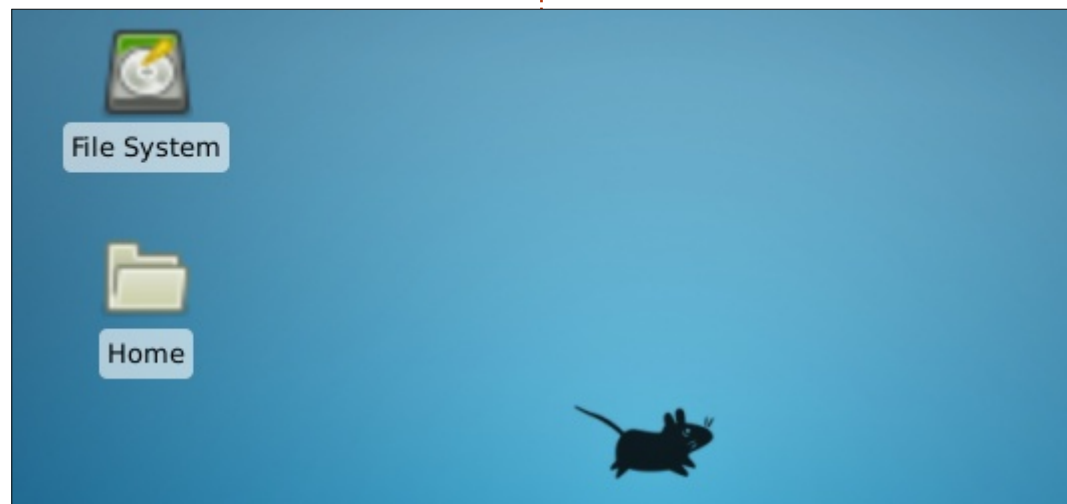
interface for the time being, until we install the graphical interface. With the df and free commands, we can see how this type of interface uses up very little disk space and memory. On my test system, the installed server system was using a mere 942MB of disk and 98MB of RAM:

INSTALLING XFCE

The first thing we will need to do is make ourselves administrator (super-user). We can either use the sudo command before each succeeding step, or use it just once to create a new shell as admin:

```
$ sudo bash
#
```

Note the change of prompt, going from '$' to '#'.

Before installing anything, we need to refresh the list of software packages from the Ubuntu repositories. Setting up a WiFi connection from the command line is slightly involved, so just connecting our machine with a physical (Ethernet) cable will be much easier. Once connected, we will allow the system several seconds to set up the network interface automatically, and then type:

```
# aptitude update
```

You should see the system connect to the repositories, and download a list of the most recent versions of each package. We can proceed by installing both the X-

Window server (graphical subsystem) and the XFCE desktop at the same time. Use:

```
# aptitude install xfce4-
session pulseaudio xinit x11-
session-utils
```

We should be requested confirmation to download some 59.7MB of data, which is then unpacked and set up. When the process has finished, the df command should report about 1.2GB of disk space used.

We are at the end of the first step. At this point we should have the desktop session installed, that we can execute manually using the startx command. This is a minimal XFCE desktop, with no panel (and

no applications) for the time being.

We should complete the installation with the XFCE panel, an icon set, and at the very least a terminal. This will take a download of just 6MB between them:

```
# aptitude install xfce4-
panel xubuntu-icon-theme
xfce4-terminal
```

We have completed the second step. We have a minimal XFCE desktop working, though it still needs to be started up with the startx command. We may need to start the panel manually the first time we use it; to do so, start up a terminal within XFCE, and type

```
$ xfce4-panel &
```

HOWTO - MINIMAL UBUNTU DESKTOP
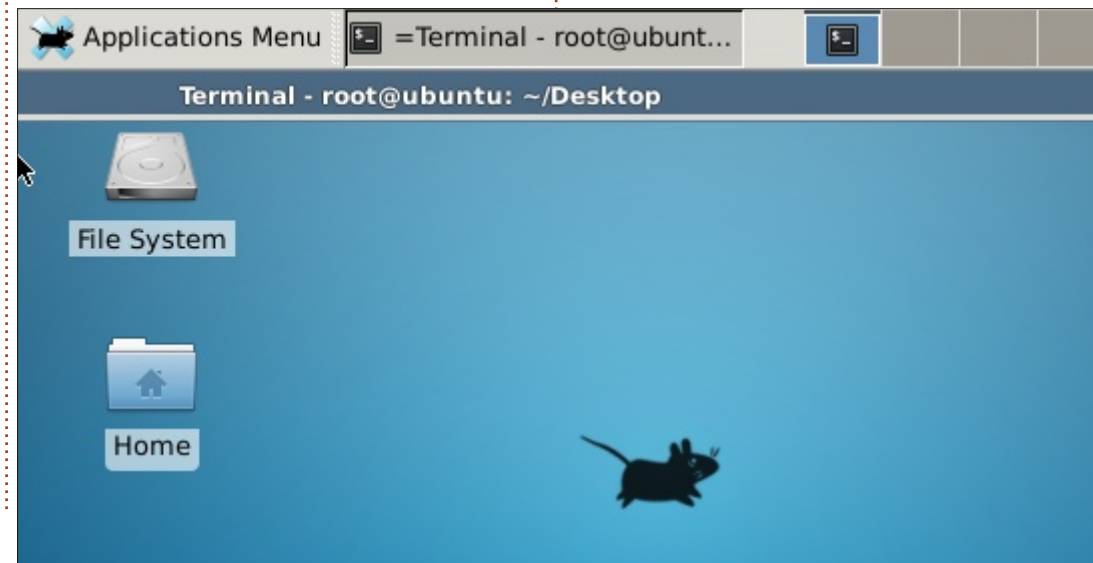
INSTALLING A DISPLAY MANAGER

Moving forward, most users would like the graphical desktop to come up automatically at boot, without having to use the startx command manually each time. A simple solution would be to write the appropriate script, but this would be a departure from the standard Ubuntu configuration. In the interest of ease of maintenance, we will install a display manager instead.

The display manager is a type of program that offers us the typical login screen of desktop installations. It has two main functions: on the one hand performing authentication and allowing access to the user's desktop, and on the other giving the user a choice of desktop session and session language.

However, most personal computers are typically used by a single person, who uses the default system language and does not often change the desktop session manager. If this is your case, it may make sense to prefer the nodm package over any of the standard options. This package starts up the desktop session directly, without presenting a login screen. A look at the various options shows us that this is the least costly choice in terms of download size, even better than the already lightweight XFCE display manager, xdm:

```
lightdm: 192 MB
kdm: 103 MB
gdm: 81.8 MB
xdm: 172 kB
nodm: 32 kB
```

We use the standard procedure to download nodm:

```
# aptitude install nodm
```

We also need to edit configuration file /etc/default/nodm, replacing line

```
NODM_ENABLED=false
```

with

```
NODM_ENABLED=true
```

In the same file, modify line

```
NODM_USER=root
```

with whatever user login you prefer.

This is the end of the third step: we have a minimal Xfce4 desktop that launches at boot.

Note that when the user logs out, the X session reboots and the user is directly logged in once more. This behavior is probably not what the more security-conscious would recommend, but may be acceptable for machines that are always physically secure.

FINISHING UP

We will probably need at least a web navigator to make effective use of this system. Installing the Firefox navigator is a simple way of transforming it into both a web access station, and a general-use office machine based on web services. An equivalent of a Chromebook, so to speak, but on your own hardware. This navigator takes a 30.7MB download:

```
# aptitude install firefox
```

Just to clean things up, we can remove the memtest package that is rarely used on a stable system, as well as the aptitude cache of downloaded software:

```
# aptitude remove memtest86+
```

```
# aptitude clean
```

By this point, disk occupation has gone up a little, to about 1.4GB. This is still less than half that of a standard Ubuntu installation - so we seem to have achieved our aim.

Under these conditions, RAM usage can go down as far a 172MB with the graphical desktop in use. It will increase, however, as soon as we start browsing the web or using other programs. Once Javascript and Flash are in action, there is not much difference in terms of RAM usage between lightweight and standard installation, though some savings can be made and may be worth it on a lightweight computer.

Alan teaches computer science at Escola Andorrana de Batxillerat (high-school). He has previously given GNU/Linux courses at the University of Andorra and GNU/Linux systems administration at the Open University of Catalunya (UOC).

It probably won't surprise you to learn that I use LibreOffice to write these articles. For the longest time, I would print out each draft of an article, mark it up with a red pen, then transfer my changes to the computer. This old-school method came from my years editing papers as a student and as a professional. As I began to write on a regular basis, I accumulated a growing stack of recycle paper from writing. I tried to use both sides of the paper, and even ripped some in quarters for use as note-paper. Before long I was burning through a lot of paper and ink.

At some point I decided I needed a digital solution. I already knew that Writer provided editing markup, but had never tried to use it for editing my documents. With some hesitation, I began to use it. The whole "old dog, new trick" issue. After using it for a couple of months, I began to wonder why I hadn't used it before. From right inside Writer, I can track my changes, make notes (comments), and accept or reject the changes. In many ways, it is quicker and even more efficient than the hard-copy method.

PREPARING TO RECORD CHANGES

Once I finish the first draft of my article, I save it as a version and turn on recording changes. (More on version control later.) 'Edit > Changes > Record' sets Writer to begin recording changes. Edit > Changes > Show sets Writer so it shows me the changes that have been made. I have debated with myself the wisdom of showing changes while I am actually editing. On the one hand, I can see the changes as I make them. On the other hand, showing the changes while I edit can make reading the text difficult. I have tried it both ways, and see some logic in not showing changes during the actual edit process, but waiting until the accept and reject process to actually show the changes. In the end, the choice is yours.

NOTE: If you are passing the document to someone else to edit, you might want to take some precautions. File > Properties; select the Security tab; check "Record Changes", and click on "Protect". Enter and confirm a password. This will prevent the other person from making changes and then accepting or rejecting them. When you get the document back with their changes, File > Properties, click on "Unprotect", and enter your password. You can now accept or reject their changes.

RECORDING CHANGES

If I set Writer to show changes, and I add text, it is underlined and changed to the color I have set. When I remove text, it is shown as struck-through just as I would when editing a hard copy. The strike-through text is shown in a different color from the added text. The colors used for added and deleted text are controlled by the settings at Tools > Options > LibreOffice Writer > Changes.

If I hover over a change, it will show the author, date, and time of the change in a tool-tip box. If I have "Extended tips" turned on in Tools > Options > LibreOffice > General, the tool-tip will show the author, date, time, and any comments attached to the change. The author's name is controlled by the information in Tools > Options > LibreOffice > User Data.

To add a comment to a change, I place the cursor anywhere within the changed text. Edit > Changes > Comment brings up the comment
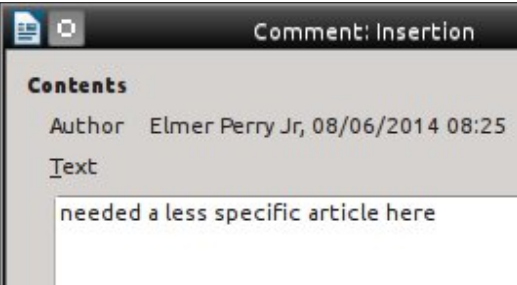
dialog. I can then enter my comment for the change. This is sometimes useful to remind myself of why I made a certain change. Once I am finished, I click on OK and the comments are added to the change.

ACCEPTING / REJECTING CHANGES

After editing the draft, I am ready to review my changes. If changes are not currently showing, Edit > Changes > Show. Edit > Changes > Accept or Reject brings up the "Accept or Reject Changes"
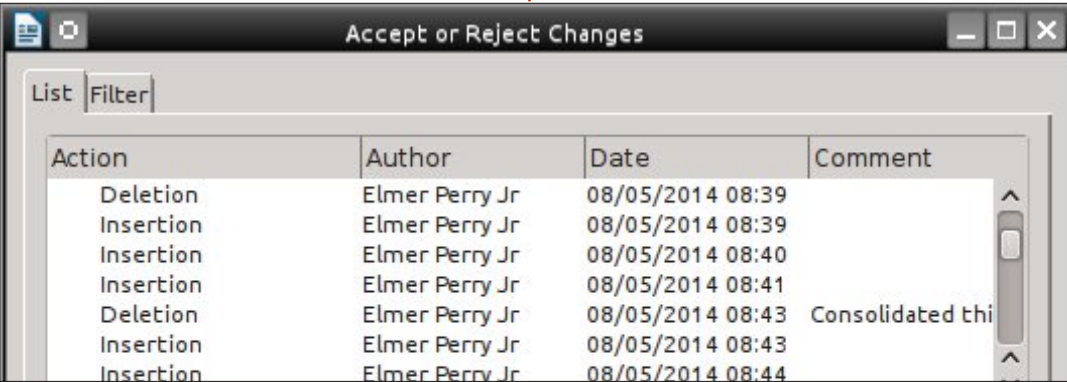
dialog, which displays a list of all my changes. Each listing shows the type of change, author, date, time, and any comments. Selecting any of the items in the list highlights the change in the document. If I want to keep the change, I click the Accept button. If I don't want to keep the change, I click the Reject button. If I know I want to keep all my changes, I can just click the Accept All button. However, if I decide to throw out all my changes, I can click the Reject All button.

On the Filter tab, I can narrow down the kind of changes that are displayed in the list. There are four different filters for me to choose from: Date, Author, Action, and Comment.

The Date filter allows me to see only changes that were created earlier than, since, equal to, and not equal to a certain date. I can

also select changes that occur between two dates. The buttons with the clock face at the end of the date entry fields allow me to set the date to the current date and time by clicking them. The last setting for the date filter is to show only the changes made since the last time the document was saved.

I never have much use for the Author filter, as I am usually the only one who is making changes. However, if I did have someone else editing for me, I could pick between my own changes and the changes made by another editor of the document.

The Action filter allows me to display just a certain type of change. The four change types are insertion, deletion, formats, and

table changes.

If I feel the need to add a lot of comments to my changes, I can use the Comments filter to filter changes based on the text contained in the comments. I just check the Comment filter and enter the text I want to look for.

MARGINAL NOTES OR COMMENTS

Sometimes, I need to make a note to myself. Using the hard copy method, I would just write a note in the margins. These usually don't represent an actual change, but an idea or action to act on at a later time. For example, needing to expand on an idea, or save an idea for a later article. Marginal notes are accomplished through
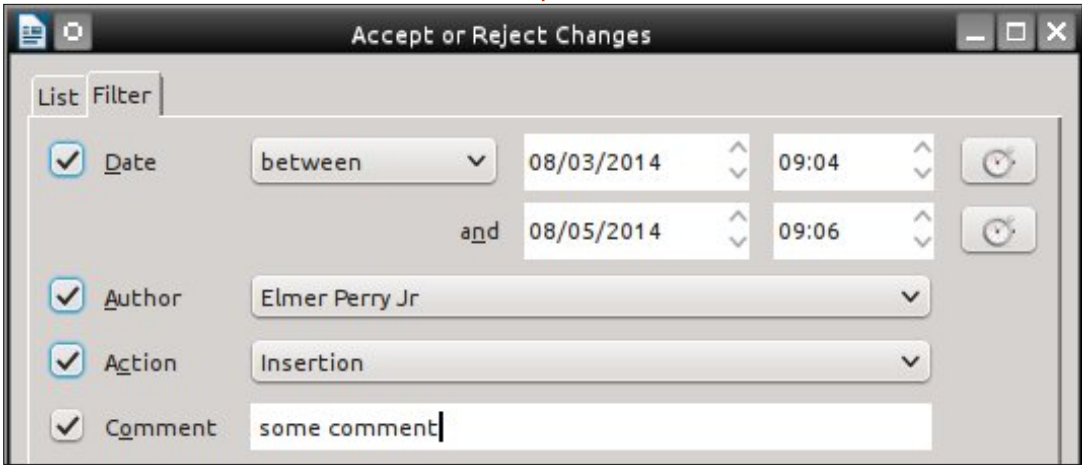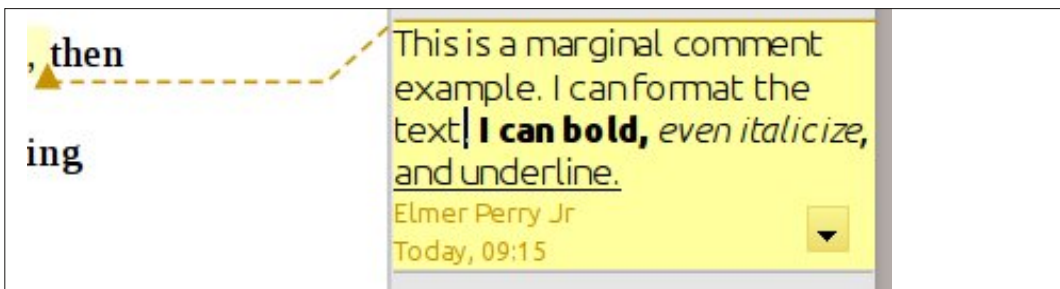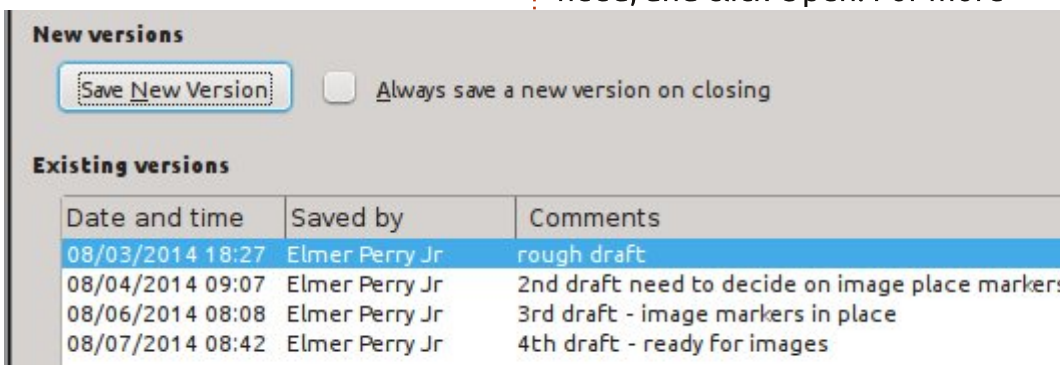
Comments. Do not confuse these with change comments.

To create a comment, I highlight the information it applies to. Insert > Comment. A box in the color assigned to the author by LibreOffice appears in the right margin. The name that appears at the bottom of the comment is controlled by the data entered in Tools > Options > LibreOffice > User Data. Placing my cursor inside the box, I type my comment. I can even format the text within the comment box using the format tools in the formatting toolbar or the side-bar. I right-click or click on the down-arrow to get a menu that allows me to delete comments when they are no longer needed.

SAVING VERSIONS

After each cycle of changing and then accepting / rejecting, I like to save a version of the document in its current state. File > Versions opens the Versions dialog. I click on Save New Version and give the version a name like "Rough Draft", then maybe any comments I may need like "image markers in place." I click OK to save the version. If I ever need to go back to a version, I can open the Versions dialog, select the version I need, and click Open. For more information on version control, see my blog post at http://wp.me/pvwht-9k.

NOTE: The version control in LibreOffice is okay for small to moderate size documents, but less than ideal for very large documents.

The editing tools built into Writer are useful for single-person editing or multi-person editing. I can have the program track my changes and allow me to later accept or reject the changes. I can add comments to the changes to help me remember why I made the change. I can also add margin notes to the document through the comment tool. After each edit, I can save a version of the document in case I decide to revert back to a previous version.

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

In previous parts, I explained how GRUB (version 2) works and how you can tune its behaviour, either instantly through the GRUB terminal or permanently by editing a couple of files and running update-grub. Now we arrive at the final stage of this exciting adventure! We will customize GRUB so that it offers a choice between multiple installed operating systems (OSs) to choose from. We will do it in such a way that this GRUB menu no longer depends on one of the installed distros, but is at your command only…

Before starting, we need to know some basics about partitioning. One physical hard drive can be 'divided' into separate partitions, which are formatted according to a specific filesystem (like FAT, NTFS, ext3, etc. – refer to Wikipedia and other sources if you are unfamiliar with this). When installing multiple (Linux) operating systems, you will need a number of partitions, plus one for swap space if it is needed on your machine (swap serves as extended,

but much slower, memory on the hard drive; see online resources). Some may want to take a separate /home partition into account. However, better avoid having several Linux installations use the same /home partition, as that may lead to inconsistencies.

Is there a (relevant) limit to the number of partitions on one hard drive? Well, it depends on the type of partition:

• one physical hard drive can hold four primary partitions at most. This is the basic (and default) type of partition;
• if you need more partitions (like us), you have to do this with an extended partition. One hard drive can hold one extended partition at most, and it counts as one of the maximum four primary partitions on the hard drive. So if you already had four primary partitions, you have to delete a primary partition first and create an extended partition instead;
• one extended partition can hold as many logical partitions as you like (including swap). They can

be formatted as any filesystem (while an extended partition cannot be formatted: its only purpose is to hold the logical partitions).

In Linux, the partitions are denoted as /dev/sda1, /dev/sda2, /dev/sda3, etc. (considering partitions on your first hard drive, i.e. /dev/sda). To access their contents, you may have to mount each one as follows (for the first partition):

```
mkdir /mnt/partition1 && sudo
mount /dev/sda1
/mnt/partition1
```

That is, if they are not already mounted automatically by the distribution you are working in.

An excellent way to create-extend-move-shrink-delete partitions, is to use GParted in a live session from USB-stick/DVD (that will allow you to modify everything on the hard disk). The program is quite self-explaining, but clear guidance can be found on the internet. If GParted is not yet installed in your downloaded iso-

file, it is only an 'apt-get install gparted' away. After having made the desired partitioning layout on the hard disk, it is always a good idea to write down the partition numbers (know what /dev/sda1, /dev/sda2 etc. correspond to), as well as the unique identifiers (UUID), which you will find in the properties of each partition. Also, Disk Utility (palimpsest) from the gnome-disk-utility package provides a nice graphical overview of your partitions. Finally, commands like 'sudo fdisk -l', 'sudo blkid' and 'sudo parted -l' will tell you similar things.

The rest of my story is to give you one (of several possible) ways to achieve a multiboot system with an independent GRUB on the MBR of your hard drive. The key will be to provide one extra partition on which /boot/grub - and nothing else - resides with all its usual contents (including grub.cfg). Let's call this partition GRUBpart. ONLY the GRUB installed on the Master Boot Record (which I will call MasterGRUB) is allowed to refer to this partition! The grub.cfg file on

GRUBpart should only contain menu entries that chainload either Windows or GRUB installations on the distribution-specific partitions. The GRUBs on the Linux partitions use their 'own' /boot/grub directory within the distro's partition. Therefore the /boot/grub folder on GRUBpart will NOT be used or updated by any of the installed distros! Consequently, when you replace one distro with another, everything still works: the MasterGRUB just continues to chainload the partition-specific bootloaders. Of course, during installation of a new Linux distro (replacing a previous one), you will have to specify that GRUB must be installed on the specific partition (/dev/sda4 for example), and NOT on /dev/sda (which refers to the entire disk, and therefore to its Master Boot Record).
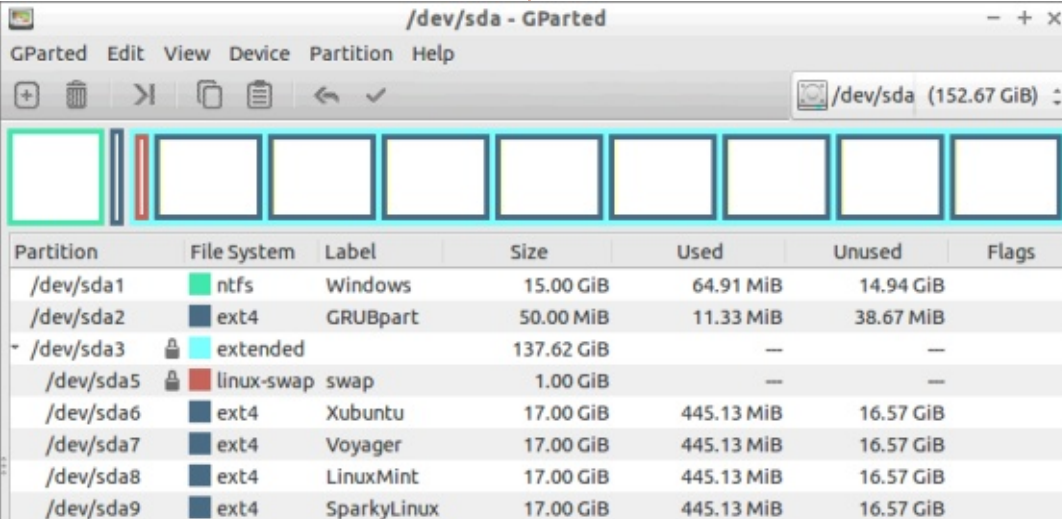
Let's make such a system in 10 steps. But be prepared first! Especially by learning GRUB terminal commands and knowing how GRUB works; see previous sessions for more information. I have tested the below myself, but of course I am not responsible for your own GRUB- and PC-adventures…

I will start from a blank hard disk, so if you already have something installed, first read everything in order to decide what you will need to do. And in that case: first backup all your precious stuff please… To be broadly applicable, I presume that you will install Windows as well – but of course that is up to you.

• Use GParted from a live session to create two primary partitions – Windows and GRUBpart – and one extended partition with the number of logical partitions therein corresponding to swap (option) plus the (maximum) number of Linux distros you may want to multiboot. For Windows, use the NTFS file system, and for Linux and GRUBpart use ext2 or higher. Of course you have to decide on partition sizes yourself; I consider 15 GB for the OS partitions as a conveni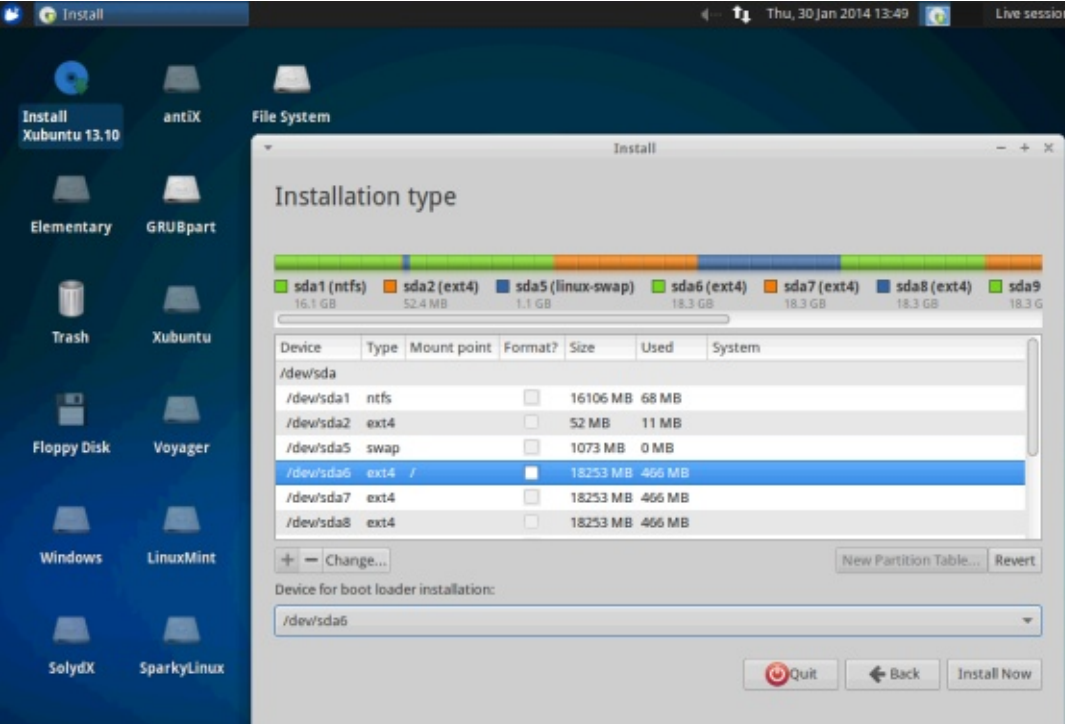ent lower limit. Decide on whether you need a partition for swap (format it as such), and on its size. If you only have 1 GB RAM, it is wise to add another GB as swap. Use at least 50 MB (yes, just mega-) for the GRUBpart partition.

• Install Windows onto its primary partition. Beware that creating an NTFS partition beforehand (step 1) was necessary to prevent Windows 7 (and alike) to create an extra primary partition for specific system files. It would do so when you installed Windows on an unpartitioned hard disk. And in that case, live with it and shrink partitions (see link at the end).

Keep in mind which Linux distro you (eventually) want to let install its GRUB on the MBR. My choice would be a distro that holds the latest version of the grub-pc package (see distrowatch.com for an easy overview). Let's call that distro 'distro Z'. Install each Linux distro, except Z, on its own partition. For all distros except Z, install their GRUB (during installation) on their OWN partition; this differs from the suggested default during installation. This means that you will usually have to change /dev/sda as GRUB's target into - for example - /dev/sda5 (correctly choosing the partition number of course).

• Install distro Z as the last one. Here we will follow the default /dev/sda to install GRUB to. Now we get GRUB in the MBR pointing at /boot/grub on the Z partition. This gives us a bootable machine, which is the reason to take this way (a more advanced alternative is that you handle Z like in step 3, and that you accomplish everything else from a live Z session).

• Boot from your hard disk, and check that all OSs are able to boot from the GRUB menu that Z installed. If not, figure out what went wrong and get it working.

• Boot into distro Z and install Z's GRUB onto its own partition (supposing partition 8):

```
sudo grub-install /dev/sda8
```

You may get some warnings, but the job should get done (otherwise, add --force). At this point, both MasterGRUB and Z's GRUB use the same grub.cfg.

• In distro Z, add chainload entries to /etc/grub.d/40_custom (see previous time) that chainload the respective OSs (including GRUB on partition Z itself – indeed, you will be able to loop, for now). Tune /etc/default/grub according to your taste; you may want to add a background image in /boot/grub (check our previous session!). Now run:
```
sudo update-grub
```

Inspect the updated grub.cfg for abnormal stuff, and reboot if OK. Try out all your chainloaded GRUBs. We are nearly there. Remember that it is still distro Z that is governing the MasterGRUB.

• Boot into distro Z. We will now overwrite GRUB in the MBR with one that points at the GRUBpart partition. We will also make two different grub.cfg files, one for MasterGRUB (containing only the chainload entries), and one for distro Z. Let's do this in a root terminal from distro Z:
```
sudo -s
```
```
cd /boot/grub
```
```
cp grub.cfg grubZ.cfg
```

here we save Z's current grub.cfg as a backup
```
cd /etc/grub.d
```
```
chmod -x *linux* *lupin* *memtest* *prober*
```

don't mind if one of the patterns is not available
```
update-grub
```

this creates a grub.cfg which we can use for our MasterGRUB, with the chainloaded entries only
```
chmod +x *linux* *lupin* *memtest*
```

to have future grub-updates in Z making a "normal" grub.cfg for Z again —- # don't mind if one of the patterns is not available
```
mkdir /mnt/GRUBpart && mount /dev/sda2 /mnt/GRUBpart
```

of course replace sda2 with the appropriate number of the GRUBpart partition. You can skip this step if GRUBpart was already mounted.
```
mkdir /mnt/GRUBpart/boot
```
```
grub-install /dev/sda --boot-directory=/mnt/GRUBpart/boot/
```

you may get some warnings, but the job should get done (otherwise, add --force).

Yes, we just installed our final MasterGRUB, which is told to look into the GRUBpart partition!

For version 1.98 and earlier, set —-root-directory=/mnt/GRUBpart/ instead of the —-boot-directory option.
```
cd /boot/grub
```
```
cp grub.cfg /mnt/GRUBpart/boot/grub/
```

and do the same for a possible background image if grub.cfg pointed at one
```
mv grubZ.cfg grub.cfg
```

i.e. we change its name back to restore Z's grub.cfg
```
exit
```

To finish grub.cfg in the GRUBpart partition, you need to look for all lines in grub.cfg above your custom entries and replace pointers to the Z's partition with pointers to GRUBpart. In our example this means replacing hd0,8 by hd0,2 and also the corresponding listed UUID's, which you get with 'sudo blkid'. This is necessary because grub.cfg points to some other files and modules in /boot/grub (such as the background image), and we want to specify that is to be found on GRUBpart.

• Check out that everything works fine now. We accomplished the essentials of our mission, except for one thing: the grub-pc package of distro Z is still configured to install future updates of GRUB onto the MBR (that is, when new versions of the grub-pc package arrive). To change its default target, run the following in distro Z:

```
sudo dpkg-reconfigure grub-pc
```

After some questions, for which you can safely accept the default, you will be asked to set the default target device to install updates of Z's GRUB, which must be its own partition ONLY (/dev/sda8 in our example). Select it with an asterisk by hitting the spacebar. Also grub.cfg (in /boot/grub on /dev/sda8) will be overwritten in this process.

• Tune each distro's own grub configuration file according to your taste (edit /etc/default/grub plus contents of /etc/grub.d/, and run 'sudo update-grub'). Among other things, you will want to get rid of the chainload entries of 40_custom in distro Z. A tweak that I like to add in 40_custom of each distro is a chainload entry 'Go Back' that will redirect the user to the MasterGRUB menu (use set root='(hd0)'). Just in case you had picked the wrong distro – or changed your mind!

Well that's it! It may have taken some time to learn things, but after all it was a relatively easy job. You are now the proud user of a multi-OS machine with a boot loader that is no longer controlled by any of the operating systems. This also means that you need to manually edit grub.cfg on the GRUBpart partition when necessary, i.e. typically by changing a menu title or adding a menu entry – but that is easy now. For future Linux installs, just keep in mind to install each distro's GRUB on its own partition. Your MasterGRUB chainloads them. Happy booting!

Floris is very grateful to **David H.** and **Bart A.** for their useful comments on an earlier version of the manuscript.

INTERESTING LINKS

Ubuntu Community 'swap FAQ':
https://help.ubuntu.com/community/SwapFaq

Ubuntu Community 'How to partition':
https://help.ubuntu.com/community/HowtoPartition

Resize Windows partitions with Linux:
https://help.ubuntu.com/community/HowtoResizeWindowsPartitions

Dedoimedo GRUB2 full tutorial, on GRUB configuration and multibooting:
http://www.dedoimedo.com/computers/grub-2.html

Sanket Totewar's article on GRUB2 multibooting: to download from
http://www.linuxidentity.com/us/index.php?name=News&file=article&sid=5047

Floris Vanderhaeghe became a Linux fan through Ubuntu 10.10. You can email him at tux7546@gmail.com.
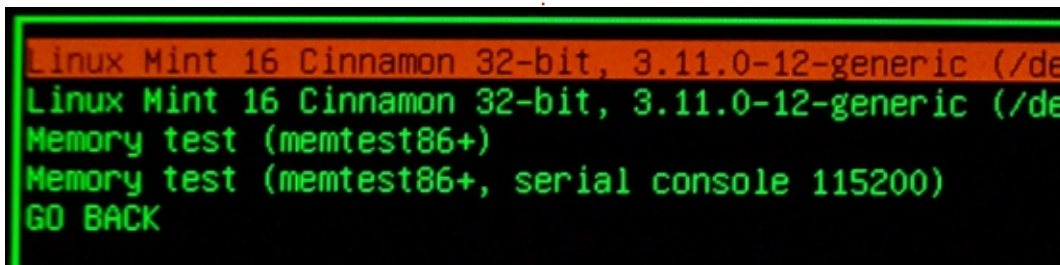
It's Render time! We have encountered render already (by pressing F12 to render an image) in some previous FCM issues, but now we are going to render an animation.

Press the render Tab (right) from the Properties window.

First of all, there are 3 choices under Render context: Render (renders a single image), Animation (renders the entire animation), and Audio (renders a sound file of your animation).

We're interested in the second choice (Animation) as we want to render the entire movie. But before pressing this button we have to inspect some things first.

A good question to ask yourself before rendering something is "what is the next use of my finished file".

If, for example, you work in the film-making industry as I do, maybe you have to render your animation as an image sequence so that every
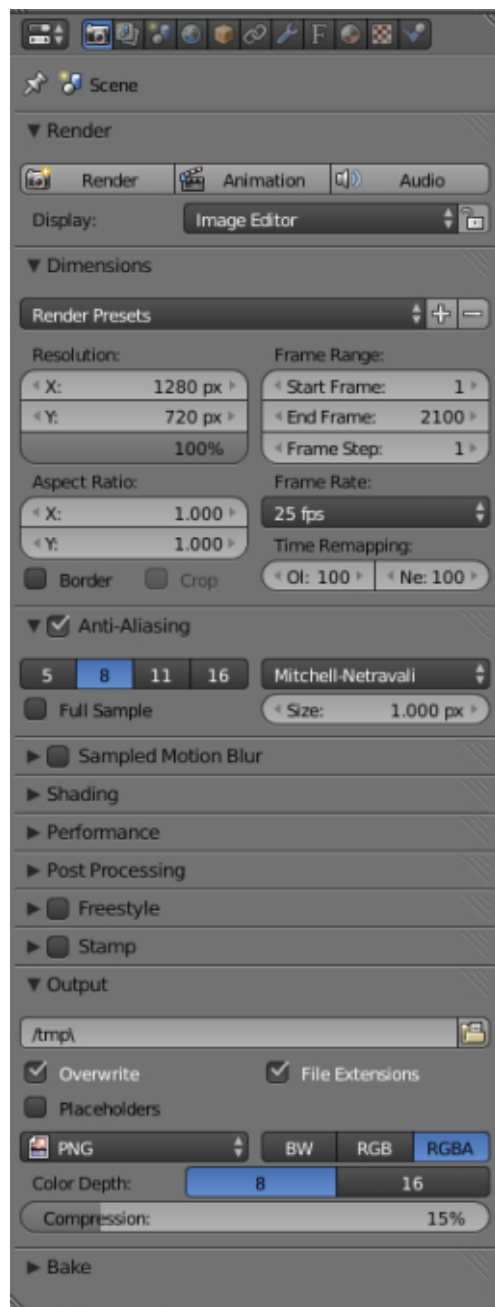
other co-worker can have the image sequence ready for importing to his suite (avid, sony vegas, final cut or premiere for editing, flame or nuke for compositing, etc.), and they don't have to worry about codecs.

If you want your rendered file to be ready for youtube or vimeo, you have to render it as a mp4, a quicktime or a mpeg2 file – youtube and vimeo accept other video files too.

Lets assume that we want a video file to show our creation to our friends.

A quicktime file with an H.264 codec will do the job.

Now, let's move on to the next subtext called Dimensions. Here we can adjust the resolution of our exported animation, the frame range that we want to render (in case we want to render just a portion of the entire animation), next we can adjust the Aspect Ratio, the Frame rate (for Europe it's 25fps), if we want to add

borders in our render we have to tick Border, and Time Remapping where we can adjust the time, making the entire animation play slower or faster.

I suggest we use some of the presets under the Render Presets pull-down menu.
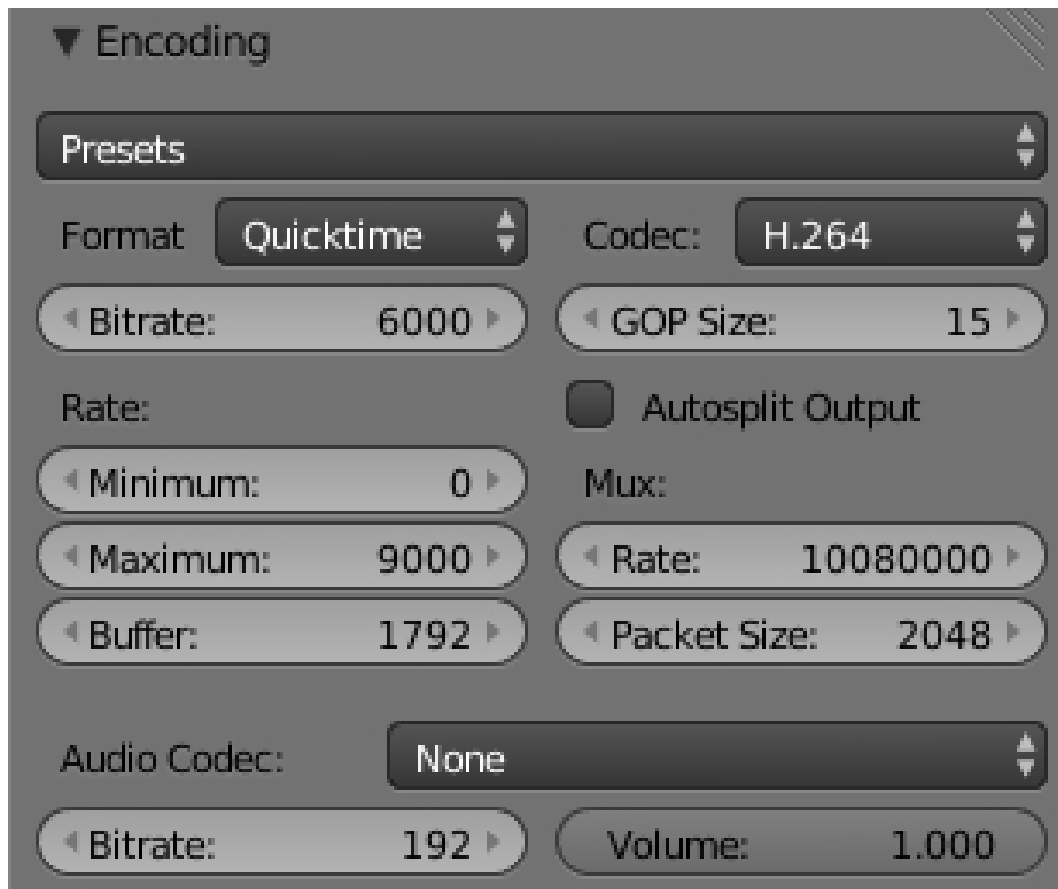
Leave all the other settings to their defaults except the Output subtext. This is where we choose where to put the rendered file or image and also what type of video or image sequence to render.

Let's select H.264 from the pull-down menu (which, by default, has PNG).

Another subtext appears called Encoding.

Here we can select Quicktime for the Format and H.264 for Codec.

Leave the other settings to their defaults. (The bitrate defines the quality of your rendering, the higher the better, but higher also

These are the most basic things to know in order to render a standard scene. The Rendering process is a whole separate study in Computer Graphics and Film Industry.

If you are interested in the Blender Game Engine, as I covered in last month's issue, check out this interesting site: http://www.tutorialsforblender3d.com/

creates larger files and takes much more time to render).

We don't have audio so we don't have to mess with that for now.

Now, we are ready to render our animation. By pressing Ctrl-F12 we start the rendering process. Remember that the rendering process is by far the heaviest user of computing power, so, depending

from the scene's weight that you are rendering, the whole rendering process will take from some minutes to hours or even days to finish! By "scene weight" I mean the number of objects, the lighting of our scene, the detail of our objects, the textures, etc.

Finally after a while (depending on your PC's power) we have the video file or the image files that we rendered.

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

**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!**

In this article, I'm going to start looking at clones in Inkscape. Clones can easily be dismissed as a minor feature, but, with a few tricks, you can use them to perform marvellous feats that make them worthy of several articles just to cover the basics.

At the simplest level, a clone is nothing more than a duplicate of an Inkscape object that maintains a link to the original. Changes to the original are automatically propagated to the clone, making them ideal when you need several identical objects in an image. Creating a clone is as simple as selecting the object and using the Edit > Clone > Create Clone menu entry, or just pressing the Alt-D keyboard shortcut. The new clone will be created directly on top of the original object, and will be selected automatically so that you can immediately move it to somewhere else.

With a clone selected, the Edit > Clone > Select Original menu item (or Shift-D shortcut) will select the original object with which the clone

is linked – its "parent" if you like. This now gives us a collection of related keyboard shortcuts that are well worth memorising to speed up your work with Inkscape:

```
Ctrl-X – Cut to clipboard
Ctrl-C – Copy to clipboard
Ctrl-V – Paste from clipboard
(at the mouse position)
Ctrl-Alt-V – Paste In Place
(at the object's position)
Ctrl-D – Duplicate (copy then
paste in place)
Alt-D – Clone (a linked
duplicate)
Shift-D – Select Original
```

Initially the Paste In Place option seems a little redundant, given that the Duplicate command appears to achieve the same result. The difference is that Paste In Place also works between layers, and even between Inkscape documents, so you can copy or cut from one drawing's layer, then paste into the same location in a different layer or a different drawing.

So much for the theory, let's create a duplicate and a clone and see the difference between them. For notational purposes, I'm going

to use a solid arrow pointing from the parent object to the duplicate, and a dashed arrow pointing from the parent object to the clone. With this notation the "Select Original" command always follows back from the tip of a dashed arrow to its source. Here's a simple parent object with a single duplicate and a single clone.
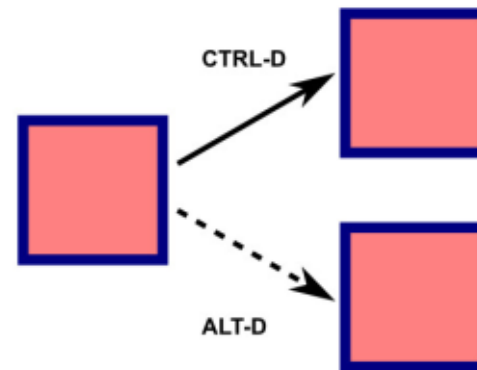

As you might expect, all three objects look the same. The difference only becomes apparent when the parent object is modified in some way – such as changing it's fill and stroke, or altering the roundedness of the corners.
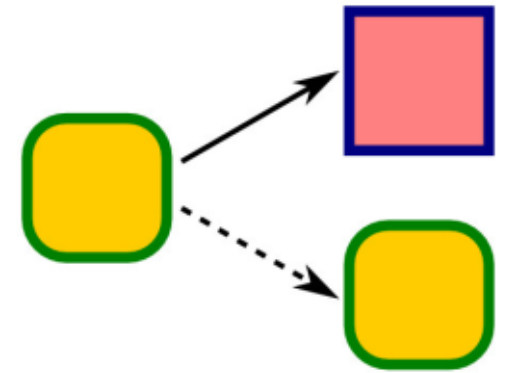
Our duplicated object, which maintains no link to the parent, remains unaffected. The clone, however, is immediately updated to reflect the changes to its parent object. That makes it useful for creating absolutely identical objects – hence the term "clones" – but, as in real life, it's possible for Inkscape's clones to develop slight differences to their parents. A clone, as well as maintaining a link to its parent, is also an object in its own right, and is subject to the same transformations as any other object. Clones can be rotated, scaled and skewed, have their opacity reduced, or have filters applied, independently of the parent. With some rotation, scaling and opacity changes applied to the

clone, our previous image looks like this.
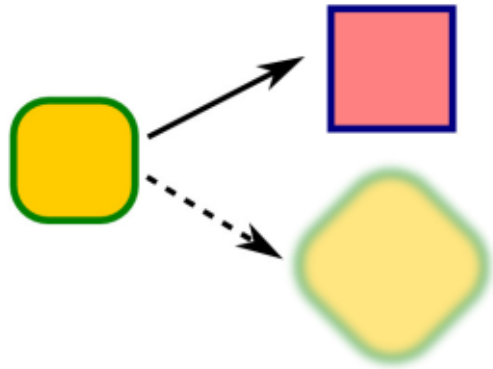
It's important to note that these changes are applied after the clone has been synchronised with its parent – you can think of them as being added on to the base attributes of the parent. So the clone's fill and stroke colors will still change if the parent's are modified, and any rotation, blur and opacity changes to the parent have a cumulative effect with those that are applied to the clone. For example, this clone has a 45° rotation applied; if I were to rotate the parent by 15° then the clone would also rotate by an additional 15° bringing its total rotation to 60° compared to its original position.

One simple but effective use of this feature is to create drop shadows by blurring a clone and reducing its opacity. This is especially beneficial when dealing with text objects as the shadow will automatically adjust to any edits you may make to the text content.

If a clone is just another type of object, then surely it must be possible to copy and paste it. If that's possible, then what about duplicating it, or even cloning it? The simple answer is that you can do all these things with clones, but the results might not always be what you expect. You can create long chains of relationships – clones of clones of clones... of clones of objects – but the cumulative nature of any applied transformations can result in a small tweak to one item having a knock-on effect all the way down the chain.

To keep things simple, I'll just look at a basic three object chain where a grandparent is duplicated or cloned to form a parent which is, in turn, duplicated or cloned to create a child. I've drawn all the possible combinations, and given each object or clone a number so that it's easier to track what's happening. Starting with the basic arrangement of objects, as with the earlier two-object example, all the elements look the same, whether they've been duplicated or cloned.
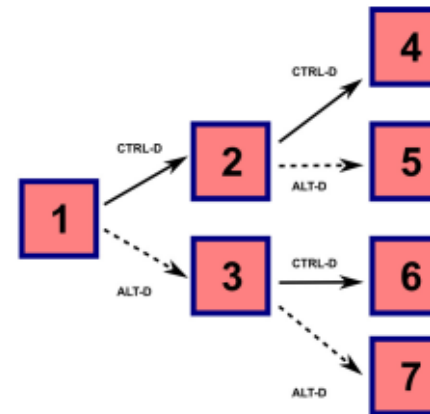

(1) is the starting point, our grandparent object. This is duplicated to create (2), and cloned to create (3). Each of those objects is duplicated and cloned to produce the remaining items. The provenance of each object is therefore:

(1) The original grandparent object.

(2) A duplicate of (1). Parent to (4) and (5)
(3) A clone of (1). Parent to (6) and (7)
(4) A dupe of a dupe of (1)
(5) A clone of a dupe of (1)
(6) A dupe of a clone of (1)
(7) A clone of a clone of (1)

What happens when we modify the attributes of (1)? As before, our duplicated object (2) remains unaffected, so (4) and (5) also appear untouched. The first-level clone (3) changes, as does its clone (7). But what about (6)? That was made as a duplicate, so you might expect it not to change, yet you can
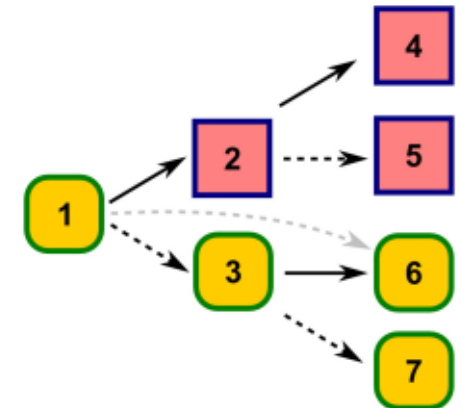

clearly see that it does.

The thing to bear in mind is that (6) is an exact duplicate of (3) – even down to its linkage to the grandparent object (1), represented by the gray line on the image. You can see, therefore, that a second clone of an object can be

created either by cloning the object a second time, or by duplicating an existing clone. The difference is that duplicating a clone will also duplicate any other attributes associated with it, so if you've rotated the clone or added a blur, those effects will be present on the new clone, even though it's linkage goes back to the grandparent object.

Now let's look at what happens if we modify (2) and (3). For this example I've rotated them through
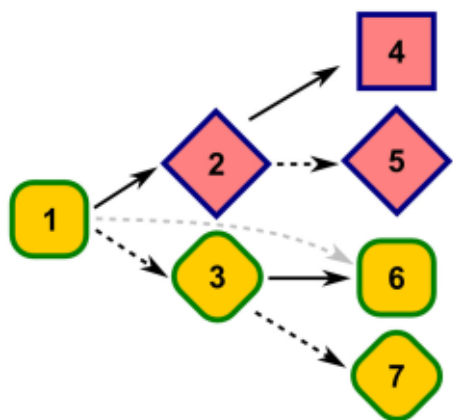
45°.

(4) remains the same, because it has no linkage to any of the other objects. (5) rotates because it's a clone of (2). The linkage between (6) and (1) means that it's not affected by the rotation of (3) – although it would be affected if (1)

were to be modified. Finally (7) does rotate because, as a clone of a clone, it's made up of the cumulative effects of all the changes made to both (1) and (3).

If that's not enough to make your head spin, I'll leave it as an exercise for the reader to consider chains of four or more objects!

One good reason for using clones is that they tend to be less resource hungry than the equivalent separate objects. In terms of storage space, a clone is little more than a reference to another part of the SVG content, helping to keep the file size down. This can be particularly useful when you're dealing with complex paths, groups or embedded images. Consider this example that uses a single image and some cloning, rather than embedding three
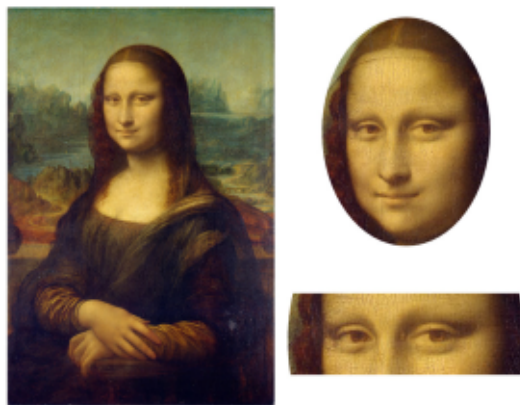
separate images.

The left hand image is the parent in this case. The top right image is the result of cloning the parent, then scaling the clone before applying a clipping path. The bottom right image is the result of cloning the clipped clone and applying a second clipping path to that. The same effect could also have been achieved by clipping a second clone of the parent, but I also wanted to demonstrate that a clone of a clipped or masked object still retains the clipped appearance – note the curved sides of the bottom right image that result from the elliptical clipping path used on the top image.

Although clones can result in substantial space savings, chaining them too deeply can slow Inkscape's rendering of your image. In the case of the Mona Lisa example, it's clear that the bottom right image can't be processed until after the top one – including its clipping path – has been calculated. Extending the chain further to create clones of clones of clones just exacerbates the problem. Unless your design really does require the sort of attribute inheritance that such chains

enable, you're better to create clones that link directly back to a common parent.

Remember, either clone from the original parent each time (Alt-D), or clone once then duplicate the clone (Ctrl-D). If you're not sure whether you're cloning the original object or an existing clone of it, check the status bar. If it reads "Clone of Clone of Clone..." then you might want to re-think your approach.

Mark uses Inkscape to create three webcomics, 'The Greys', 'Monsters, Inked' and 'Elvie', which can all be found at
http://www.peppertop.com/

This month, it's time to combine two chunks of code into one, but this time with a broader view. These two parts will form a larger beast in the coming months.

I mentioned the 16x2 LCD screen in a previous Arduino section in FCM, but I haven't mentioned the 3x4 keypad that I have. It's a bog standard, no name, 3x4 keypad, with holes in its board for wires. I've soldered in some wires of various colours and it's ready to be bolted onto the LCD screen.

I already have my LCD screen wired up to the breadboard and have the basic 'hello world' sketch running on it, so I know for sure my LCD part is right. I've previously tinkered with the keypad and monitored the output in the serial window (code shown below) but now it's time to bypass the serial window and go right to the LCD screen.

So, I have the keypad wired into the remaining digital pins and need to merge the two lots of code.

Note that I've missed out digital pin 1. For some reason when I used pin 1, I would end up with a row of keys that wouldn't output. I'm not entirely sure why. I have to assume it's to do with pin 1 being the TX, or transmit, pin.

Getting it working was pretty simple. I pasted in the include and keypad setup stuff at the start followed by editing the keypad pin numbers accordingly. In the original LCD code it printed 'hello world' in the top line then, below that, the time it'd been running in milliseconds. I replaced the 'hello world' text with 'Enter code:' and replaced the milliseconds with the key char. That was basically it.

My code is at:
http://pastebin.com/YndLneqm

Next month, we'll add more to the code, possibly more to the circuit too. Although, I'm quickly running out of pins! I have only the analog pins left.

```
#include <Keypad.h>

const byte ROWS = 4; //four rows
const byte COLS = 3; //three columns
char keys[ROWS][COLS] = {
    {'1','2','3'},
    {'4','5','6'},
    {'7','8','9'},
    {'#','0','*'}
};
byte rowPins[ROWS] = {5, 4, 3, 2}; //connect to the row
pinouts of the keypad

byte colPins[COLS] = {8, 7, 6}; //connect to the column
pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins,
ROWS, COLS );

void setup(){
    Serial.begin(9600);
}

void loop(){
    char key = keypad.getKey();
    if (key != NO_KEY){
        Serial.println(key);
    }
}
```
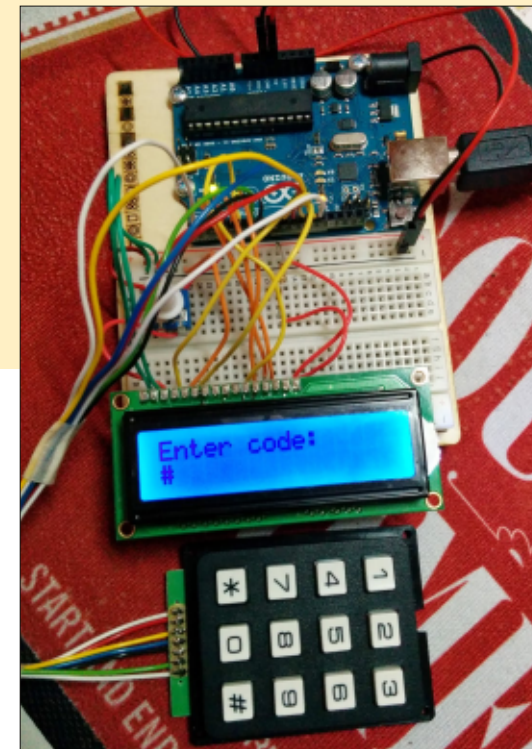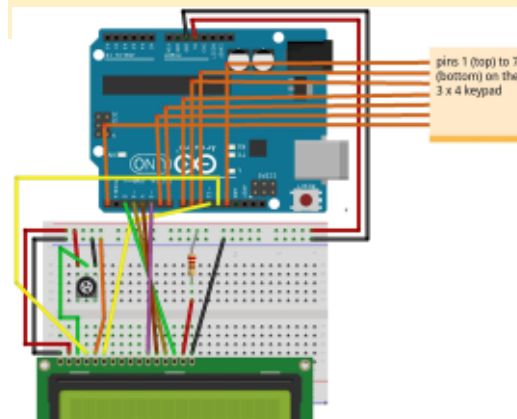
GUIDELINES

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

Ripping DVDs With Handbrake

Physical media is cumbersome. If you own a lot of DVDs, Blueray discs, VHS, or, gasp!, Discovision (circa 1978), you know how ugly it looks stored in your living room by your entertainment system. Digital media is hot for its portability between devices. Ripping DVDs is a fairly simple process and there are lots of guides around that show how to install and rip movie DVDs using Handbrake. But what about those multi-movie DVDs or DVDs with multiple episodes of a television show? Handbrake can rip those too - and the process is fairly simple.

If you're unfamiliar with Handbrake, Handbrake is, in the project's words "a tool for converting video from nearly any format to a selection of modern, widely supported codecs." Handbrake is really popular in the Mac OS X world, but is cross-platform and works on Windows and Linux. If you happen to be using the 14.04 version of Ubuntu or one of its derivatives (Xubuntu), Handbrake is in the repositories. Don't bother setting up the PPA as

shown on the Handbrake website because, as I type this, there's no PPA for Trusty (14.04). In the Xubuntu 14.04 repositories we found both the GUI and CLI (command line) versions of Handbrake. To keep things simple we installed the handbrake (GUI) package.

While you can find some good deals on particular Blueray discs, there's currently a much wider range of inexpensive DVDs available, especially when it comes to television series. We recently picked up (new) several seasons on DVD of the once-popular television show 'The X Files' for $10 each. Each season contains roughly 20 episodes (between 6 DVDs) or about $0.50/episode. The sixth DVD in each set contains extra material, so there's actually a bit more value in the set, but we tend to just rip the actual shows.

We're not huge fans of systems auto-playing media. Our fresh install of Xubuntu 14.04 was set to autoplay DVDs using parole media player. To modify this behavior we

clicked on the All Settings icon beside the lock and shutdown icons in the whisker (Xubuntu 14.04) menu, then clicked on the Removable Drives and Media icon, clicked on the Multimedia tab, unchecked the 'Play video CDs and DVDs when inserted' option box, and finally clicked close. DVDs still automount. If you want Hanbrake to launch each time a DVD is inserted just replace the parole line with:

```
handbrake --device=%d
```
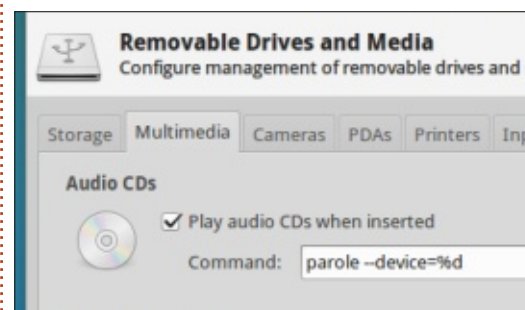

To rip multi-movie DVDs and multi-episode television shows you have to do 3 things: change the source title, change the destination file, and queue each movie or episode. Changing the source title is a matter of selecting the first,

second, third, fourth, etc. title from the drop-down source title field. If there is no drop-down, then all the titles on the DVD might be encoded as one long title (in which case handbrake would have to rip the whole thing and you'd have to manually split titles using other software). To change the title for Xfiles we used the naming convention we use for Kodi/XBMX (showname season episode):

Once you've changed the destination name, click the queue button to queue the episode/movie for ripping. Handbrake won't start ripping until you click the green encode button (which looks like a play button). Change each title and rename each destination file until you have all the episodes from the disc. Don't forget to queue the last episode. If you hit encode after changing the last title's destination, but you don't queue the title, it will not get encoded with the rest of the titles, so be sure to queue the last title too.

If you lose your place queuing
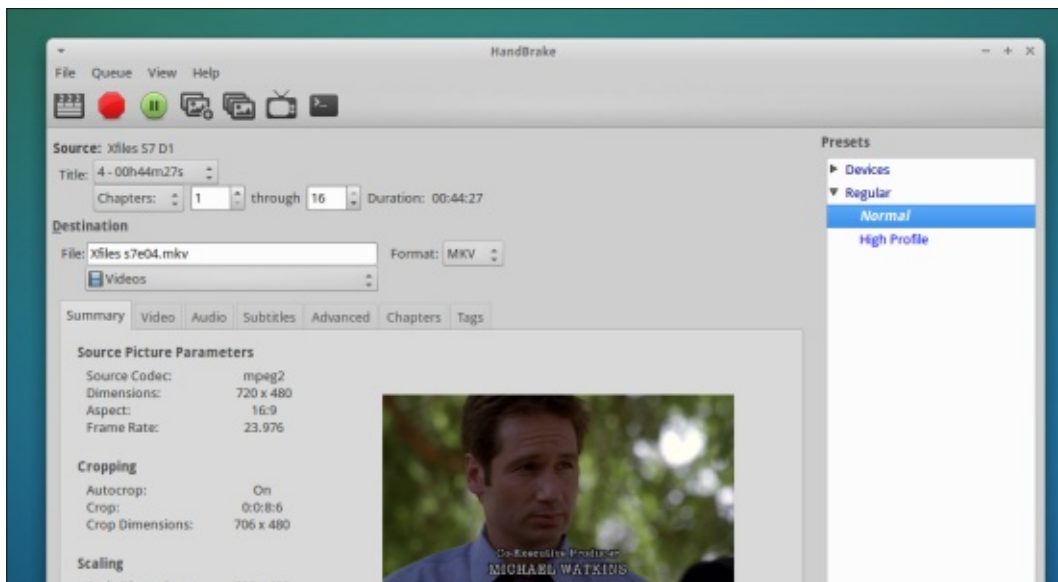
videos the bottom right of handbrake shows how many titles are pending in the queue. You can also see what's in the queue by clicking view > show queue. Show queue shows each title and the corresponding destination name. Occasionally we forget and queue two episodes with the same episode name. Removing an episode is as simple as clicking on the red x beside the queued file. From this queue screen you can also process the queue. As each title is processed there's a spinning orange icon beside the title.

We normally just process everything from the main handbrake screen, but processing from the queue screen lets you double check your work. We don't normally sit in front of the computer long. Once we start to process the queue, it's coffee time. If we sat in front of the computer for any extended period, we'd probably check the queue to double-check which titles are being processed. In general, we find we're better getting up and walking away.

On our AMD A8-5600k-based system (onboard video), it takes handbrake an average about 5 minutes to process a 45 minute episode. The average frames per second processed is anywhere between 185 to 225 frames per second. Systems with more raw CPU processing power, Intel Core i5/i7, should process faster. Of course encoding also depends on how much protection is on each DVD. Some DVDs (of similar length) take a bit longer.
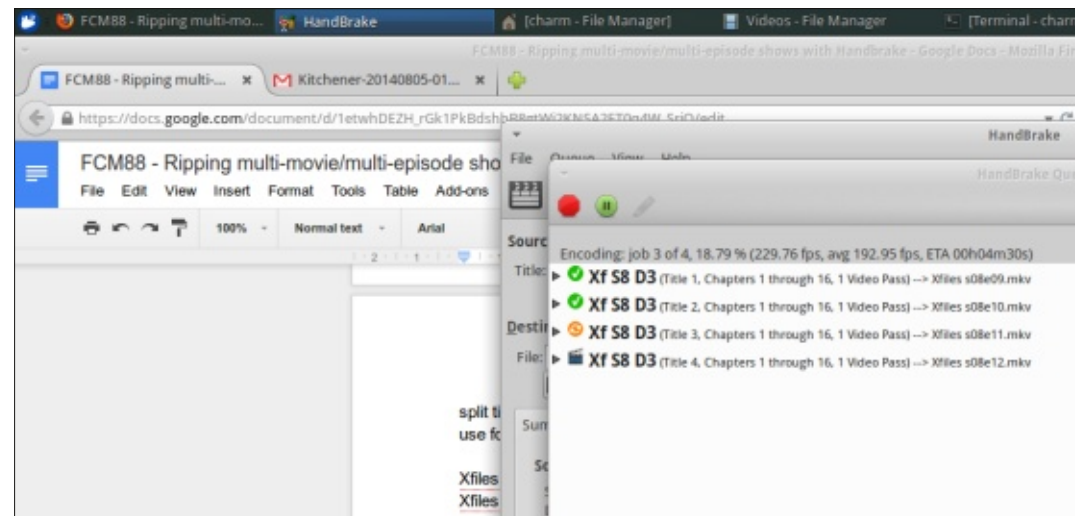
Ripping and encoding television shows from DVD isn't difficult – provided you remember to select each title, rename the destination file, and queue each title... before hitting the process button. We like handbrake because it produces nice quality files on the normal setting. Acidrip is another program worth mentioning in this space (it's available in the Ubuntu repositories); it has similar features for processing multi-episode DVDs.

Handbrake: http://handbrake.fr/

Acidrip: http://sourceforge.net/projects/acidrip/

Charles is the author of Instant XBMC, and project manager of a not-for-profit computer reuse project. When not building PCs, removing malware, and encouraging people to use GNU/Linux, Charles works on reinventing his blog at http://www.charlesmccolm.com.

Long ago and far away, compiling the kernel of a GNU/Linux system used to be what is called a rite of passage. You could not really call yourself a Linux enthusiast until you had painstakingly obtained the several different bits that form a working system - the kernel, the C compiler, perhaps also an X11 windowing system and several user programs - from different places on the Internet. The various bits and pieces would almost never work well together at first, so you needed to go through a process of compilation: i.e. transforming the source code of each program (in the C language) into an executable "binary file" - and the biggest and most complicated program was the system kernel itself.

It should also be said that in those good old days, most if not all Linux users were in fact computer people by trade or by interests.

Then came along several important steps towards making a GNU/Linux system more readily accessible to the average user. The first was the distribution, that collected all these software packages, compiled them in a coherent fashion and served them nicely bundled up in a CD image together with an installation program to make the process much more automatic. Slackware and RedHat were among the first distributions to make it to the general public, though many more came along afterwards.

A second important step that also sets the GNU/Linux world at an advantage from other operating systems - at least to my mind - is the package manager. Being able to install software packages written by different authors or projects, all directly from a common repository, definitely makes system software management easier on the administrator – be it of a single machine, and all the more so when a park of several hundred computers must be set up in the same fashion.

Finally, a third step happened with the Ubuntu distribution, when it changed the way the game was played and focused more closely on what in Apple's parlance is called the "user experience": making it not only possible but easy and even enjoyable for non-technical users to set up a working box. This is not to say that previous distributions had not obtained some progress in this direction, but Ubuntu took the process a step further, with a streamlined installation program that did not require a PhD in computer science to understand, and a large selection of world languages available straight away during the installation process itself. The importance of being able to install a system with all the messages and dialogue boxes in your own language cannot be overstated. Although most computer technicians all over the world are quite capable of understanding technical information given in English, this may or may not be the case for normal people who must contend not only with the technical barrier, but also with a language they do not always fully grasp.

At this point where we stand today, each and every GNU/Linux distribution offers at least one Linux kernel, or a main default kernel plus different optional kernels for those who need them. It has been years since most of us have actually needed to compile a kernel in anger, just to make a system work. So the question can be posed: is there really any more a valid reason for a user of a modern distribution to know how to do so?

This is the point I will try to reply to in this first part of the series. We will give some insight into what a kernel is, what it does and why it may in some cases be necessary to change it. In a second installment, we will see what we need to get in order to compile a kernel, and take a first glance at the source code itself. Further along, we will actually configure and compile a kernel, and see how the result may be installed and used on our system. We will then be able to go through some simple kernel tweaks, among which the different processor options (e.g. PAE) will be discussed. This will

bring us to performing some simple changes in the existing source code, and seeing what they do. Finally, we will go through creating some of our own code, in the form of a kernel module.

WHAT IS THE LINUX KERNEL?

On of the first diagrams that operating system students will see is the "onion" representing the different operating system layers. In this - very much simplified - version of the "onion", we see the kernel at the center of the diagram. Surrounding it, we find a layer of libraries and system utility programs. Finally, the third and outermost layer is formed by user programs.
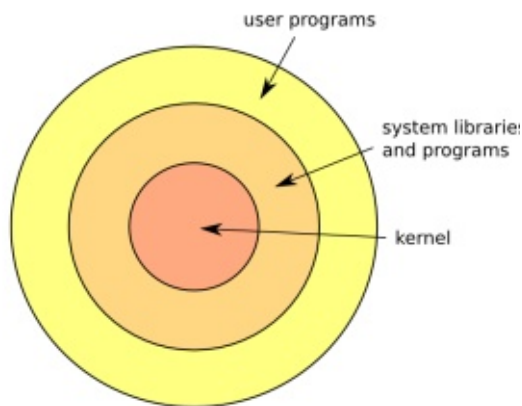
It is important to understand the purpose of each layer. The kernel itself is a very low-level piece of software; that is to say, it is in immediate relationship with the hardware and manages the most basic functionalities of the operating system. These include:

1. MANAGING THE PROCESSES AND THEIR ACCESS TO CPUS.

In a modern multiprocessing environment, computers have one or more CPUs available. Contrary to popular belief, each CPU core can in fact perform only simple tasks, and furthermore can execute only one single task in a given time-step. On the other hand, we wish to run more than one concurrent program at the same time: for example, we could very well be listening to some music with Exaile while we peruse a PDF issue of our favorite FullCircle in Evince and run an instance of Hexchat in the background. This means some part of our system needs to be available to segment each running program into short steps. Each step in then executed in turn on a CPU for a short period of time, after which it goes to sleep while other programs get access to the CPU. The process is then woken up once more and the next step is executed, and so on and so forth. The same system component that manages this will need to make sure each process gets a fair share of CPU time, that "zombie" processes are
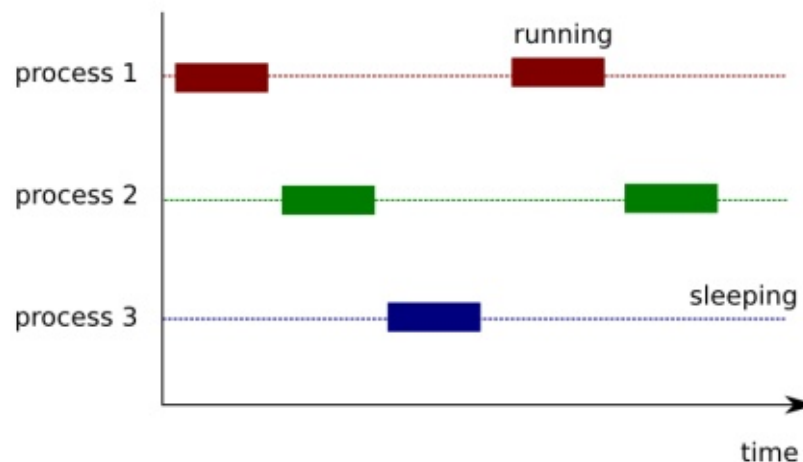
terminated, and so forth. This process management component - or "scheduler" - is part of the kernel.

2. MANAGING MEMORY.

Once more, in a multiprocessing environment, each process will, at some point, require the usage of a certain amount of direct-access core memory (RAM). If we left memory management to the processes themselves, we could expect a fair amount of competition between them: who gets access to that last page of available RAM? So we need a centralized memory management system, to which individual processes apply for access to RAM. This is also a function of the kernel, which furthermore ensures each process accesses only the memory that has been assigned to it. If it should access a page of memory assigned to another process, something visibly has gone wrong and the kernel shall immediately terminate the offending process.

3. MANAGING ACCESS TO INPUT/OUTPUT DEVICES.

Much in the same way as CPUs

and memory, hardware devices must be shared between many processes. For example, we could consider a USB port to which a printer has just been connected. Which process gets to manage this? It is the kernel that must recognize the type of driver required for this model of printer, activate it and grant it exclusive access to the USB port while the hardware remains connected.

All this can get quite involved as modern computers include new hardware types with passing time. So it is understandable that operating system kernels are just about the largest and most complex type of computer program the average user will encounter.

On the other hand, a kernel that works really well is a necessity for any computing device. Otherwise, conflicts between different running programs could not be solved, hardware would cease to be available to the software, the hard drives themselves could not be accessed, etc.

Going back to the "onion" diagram, each successive layer can request the services of layers situated inwards of themselves. System libraries and programs are respectively formed of libraries that contain much-used routines on one hand, and simple programs that any operating system would need on the other. An example of the first is the glibc library, that most if not all programs in a GNU/Linux will require. This contains many often-used routines such as writing a character string on screen, accessing a file, or writing to a network port. An example of a system program could be the mkfs.ext4 utility to format an ext4 partition. These libraries and programs will at some point need to access physical system resources, be it memory or a hardware device. At that point they will request this service from the inner kernel layer, using what is called a "system call".

This request may or may not succeed, depending on whether the requested resource is at that time available to the kernel. Certain security policies may also be in place, restricting access to resources depending on the type of program on the identity of the user on behalf of whom it is executing. In any case, the program making the request cannot directly access the resource without checking whether the kernel has granted access, although some programs have been seen to do so. The 'not taking access for granted' is one of the differences between well-written and that less well-implemented software.

Many libraries and system programs will be needed on all computers running a given version of the operating system.

Continuing outwards in the diagram, we find the user programs. These may vary from installation to installation, depending on the specific use given to the system. They will also require the service of the layers inwards of them, both of the kernel itself and also of the system libraries. For example, a web browser will request some free memory from the kernel when it starts up, in which to store pages accessed on the Internet. But if the user should access a web page through the encrypted HTTPS protocol, the browser will also require the services of the openssl

GNU, Apache, BSD and many others

GNU

Linux
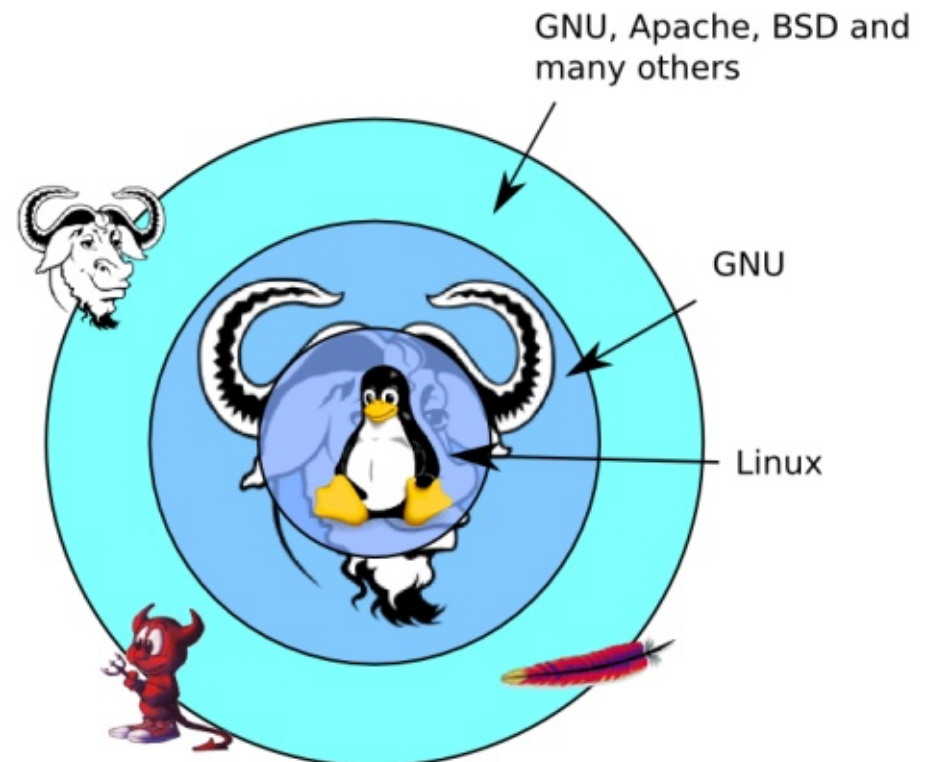
library and its routines to set up a secure channel to the server – to encrypt and decrypt data.

This explains many of the package dependencies that arise when installing new software: the maintainers of the web browser will have introduced a dependency on the openssl package, to make sure that the openssl is installed and with an appropriate version number when the browser fires up a HTTPS connection.

Some readers may have noticed that purists - such as myself - tend to refer to our operating system as the "GNU/Linux" system, instead of the more abbreviated "Linux". This is the terminology used by the Free Software Foundation and the Debian Project, among others. It recognizes the fact that in our operating system the kernel is developed by one project, started by Linus Torvalds and hosted at www.kernel.org. This kernel is in fact the only part of the system that can be called "Linux".

On the other hand, some of the most important bits of the operating system software have been developed in conjunction with the GNU Project at www.gnu.org, which is now sponsored by the Free Software Foundation (FSF). This includes the C language compiler, gcc. The GNU Project also has its own kernel, the GNU Hurd, which is quite different from the Linux kernel, and in some respects perhaps more advanced. So, by combining different kernels and keeping the rest of the operating system software, we can obtain our well-known GNU/Linux, but also GNU/FreeBSD with the FreeBSD kernel or GNU/Hurd that combines the GNU system software with the also GNU Hurd kernel.
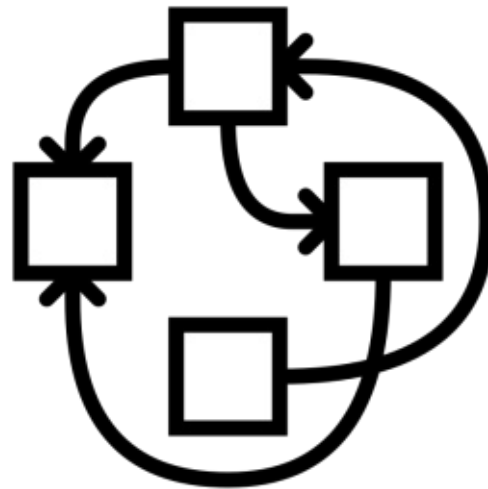
What does not help clarify the situation is that, for some time now, the Linux kernel is also released under the very same GNU General Public License (GPL) as the GNU Project software. So let us just take away that the kernel and the accompanying software of the GNU/Linux are released by different teams and leave it at that. Needless to say, many user programs have been developed in further projects, unrelated either to the Linux project or to GNU. Their software may be released under GPL, or other licenses such as the Apache license, the BSD license, or others - even commercial licenses.

WHY COMPILE YOUR OWN KERNEL?

Now that we know what the Linux kernel is, we can discuss a bit why it could be interesting for the user of a modern system to compile his or her very own kernel.

There are several reasons for this. A first point that must be made is that not all processors are equal. If we stay within the Intel product line, basically we can find two different CPU families. The first is based on the 80386 (or "i386") model released in 1985. This was a 32-bit processor, meaning that numerical operations could be executed on operands 32-bits in length. It also means that memory addresses could use 32-bits, so each process could "address" (use) up to 2^{32} address locations. This translates to a memory space of up to 4GBytes, which seemed extraordinarily large at the time.

Over the years, succeeding derivatives of the i386 (the i486, Pentium, Pentium Pro, Pentium II and III, the Pentium IV and finally the Atom) incorporated more and more features. However, these processors of the "Intel Architecture, 32-bits" or IA32 family maintained the backwards compatibility of their instruction set. That is to say that the i486, for example, introduced new functionality compared to the i386, and the corresponding new instructions were added. However, it could understand perfectly all the i386's instructions, so a program compiled for the i386 would use just the i386's instruction set, and work on both processors – just a mite faster on the i486.

This backwards compatibility was also maintained when AMD developed the 64-bit architecture

that is now used in 64-bit personal computer systems. Processors include both AMD's own line of processors, but also Intel's Core Duo, Core i3, i5 and i7 offerings. These can work in 32-bit mode, just like a 32-bit processor – which explains why, for example, the 32-bit Windows XP could be used until recently with modern processors. However, in order to take advantage of the 64-bit instruction set, we need to compile our programs and kernel explicitly for this architecture. They will then be able to execute numerical operations on operands that are 64-bits long, and use longer memory addresses to access larger memory sizes.

GNU/Linux distributions contain kernels that are compiled for a certain model of processor. Nowadays, most 32-bit kernels are compiled using the "i686" instruction set of the Pentium Pro CPU model.

At the time of writing, the two kernel packages available for Ubuntu 14.04 are: http://archive.ubuntu.com/ubuntu/pool/main/l/linux/md-modules-3.13.0-31-generic-di_3.13.0-31.55_i386.udeb for the IA32 i686

instruction set – even though the "i386" indication has been maintained in the package names;

http://archive.ubuntu.com/ubuntu/pool/main/l/linux/md-modules-3.13.0-31-generic-di_3.13.0-31.55_amd64.udeb for the amd64 (also known as x86-64) instruction set.

This means two things:
• a i686 kernel will not work on earlier models, since a i386, i486 or Pentium lack some of the instructions used. This will either simply not work at all, or may freeze during execution.
• a i686 kernel will work on later models, but will not be optimized since the more recent instructions available on an Atom processor (that was released in 2008) will not be used by the kernel.

An example of this is the famous "Physical Address Extension" (PAE) instruction set. This extension of the original IA32 instruction set allowed processors to connect to and use larger memory address sizes than the 32-bit limit set by the i386.

Originally presented in the Pentium Pro generation of Intel

processors, PAE became standard in most Pentium-III desktop and all Pentium-IV and Core series. This should include most personal computers that have been sold during the last ten years. So most people will not need to worry if our favorite distribution (Ubuntu) is activating PAE by default in its kernels since version 12.10, and so making PAE presence mandatory in the CPU. Ubuntu 14.04 will no longer work on processors without it, though other (earlier) distributions may run.

Even if we exclude users of really old hardware, a certain class of laptop that is still current in terms of usability suffers from the lack of PAE. Laptops built on Intel's Pentium M ("M" for "Mobile") processors present several advantages on later Pentium IV, M

or Core CPUs. This processor class is based on the Pentium III, which is known to be internally less complex that later Pentium IVs. In practice, they compute faster when run at the same clock speed, and so are more energy efficient and manage laptop battery life better.

So it makes sense for owners of computers, such as the original eeePC or some of the first 17" laptops, to try and keep them going - especially since, with a lightweight distribution such as Lubuntu or Xubuntu, they are still well up to most normal web browsing or office tasks.

Several solutions can be found on the web, for example those described in "Enabling PAE" (https://help.ubuntu.com/community/EnablingPAE) and "Lubuntu-

Pentium M laptops

Many Pentium M CPUs have PAE disabled by default, but it can be enabled by the user. At the moment this requires some workarounds to install or upgrade Ubuntu 14.04.

Upgrading to 14.04 on Pentium M laptop

The upgrade will be blocked with the message "Your system uses a CPU that does not have PAE enabled. Ubuntu only supports non-PAE systems up to Ubuntu 12.04. To upgrade to a later version of Ubuntu, you must enable PAE (if this is possible)". There are two things you must do to successfully upgrade:

1. **Install a PAE kernel (to verify your system is PAE capable)**

```
apt-get install linux-image-generic-pae
# reboot and then...
uname -a
# will output something like "linux ubuntu 3.11.0-17-generic ..."
# kernel version should be 3.11.0 or above (3.2.0 was the last non-PAE kernel, so anything above
that is PAE)
```

2. **Add "pae" flag to /proc/cpuinfo**

fake-PAE"
(https://help.ubuntu.com/communi
ty/Lubuntu-fake-PAE) on the
Ubuntu community documentation
server. However, simply compiling
and using a kernel with PAE
disabled solves the problem once
and for all.

The same could be said for
earlier processors. The Debian
project still supported i386 kernel
until recently, though the new base
line is the i486 instruction set (see
http://www.debian.org/releases/sa
rge/i386/release-notes/ch-
upgrading.en.html). It makes sense
for distribution maintainers to
concentrate their efforts on newer
architectures that are more
abundant in actual use, even
though this means support for
earlier models will slowly but
surely disappear. In this case, too,
running a recent distribution on
older machines will imply even
more often compiling your own
kernel.

As for more recent machines,
there are also arguments in favor
of compiling your own kernel. The
standard i686 kernel will work
quite well on modern hardware,
but will not be able to use the
more recent architecture

developments. This is the point of
view of the Gentoo distribution,
that allows the user to compile
each and every software package
installed
(http://wiki.gentoo.org/wiki/FAQ),
leading to a globally more efficient
and leaner installation.

Even if we do not need a
complete new kernel, in some
cases, when a user needs to use
relatively new hardware, it
becomes necessary to compile at
least the relevant driver. Graphics
controllers and wireless
communication devices are among
the potential candidates. The new
driver is a modular part of the
kernel, that plugs into the existing
kernel to give it the capabilities to
handle the hardware.

And finally, perhaps the best
reason for compiling a kernel is
simply because it can be done. Few
other mainstream operating
system users can say they have
compiled a main part of their
systems – but we can. Along the
way, we will also be learning a lot
about how our computer and its
software actually work.

THE NECESSARY HARDWARE

In the next few episodes, we
will be going through the process
of first obtaining the source code,
and then actually compiling and
installing a kernel. I will be using a
fresh installation of Ubuntu 14.04
on a Core i5 laptop to perform
example operations. The reader is
encouraged to go ahead and do
the same. However, the usual
caveats apply: installing a new
kernel is a major operation on your
system. Though things should
usually go well, there is some
potential for breaking stuff, and
needing to reinstall the system
from scratch. So this is definitely a
process that you should not do on
a production machine.

On the other hand, compiling a
kernel will need some raw CPU
power. Though it should be
possible on a low consumption
processor (such as a small
netbook), it will benefit greatly
from a heavyweight laptop or
desktop CPU. An Intel Core Duo,
Core i3 or similar is probably the
slowest processor that could be
recommended for this purpose.
You should also be aware that the

source code itself and the kernel
files once compiled can take up to
20GB of disk space (mostly in the
/usr directory), and plan
accordingly.

Whatever route you choose to
go, please do make sure your own
data is backed up before
proceeding.

Alan teaches computer science at
Escola Andorrana de Batxillerat
(high-school). He has previously
given GNU/Linux courses at the
University of Andorra and GNU/Linux
systems administration at the Open
University of Catalunya (UOC).

MY STORY

Written by Jacob Simon Areickal

Migration From Windows XP

I was using Windows XP on a Dell laptop with the following configuration:

CPU SPEED: 1600 Mhz
RAM: 512 MB
CPU:Pentium III

A few months back, viruses infected the computer and it became increasingly difficult to use it. The only option was a total reformat and reinstallation of Windows XP (recently orphaned). I was faced with two options, the first being reinstallation of Windows XP, and the more adventurous option of trying a Linux Distro. I was a bit apprehensive about installing Linux because of total lack of knowledge on the subject. I thus decided to play it safe and started off with the 'Run from the CD' distro called OpenSuse 12.2. The result was very discouraging because the computer was very slow and the GUI seemed very primitive.

Feeling very desperate, I tried another distro called Knoxpix, which was equally slow. From that,

I migrated to Ubuntu 12.xx which also proved to be too slow on my laptop. After browsing the net, I heard about Lubuntu 14.04 and ran it from my pen-drive side by side with Windows XP.

Lubuntu proved to be the ideal distro for this old laptop. Now,I have erased Windows XP entirely and the laptop runs only Lubuntu. After installation, there were a few challenges:

• Getting the mobile broadband mobile to work – this was ultimately solved with the help from the Ubuntu Forum.

• Installation of HP printer – this was solved with the installation of HP printing service.

Later, I slowly installed Cairo Dock and other software that I normally used on Windows. I find my laptop to be stable and quite fast; it's basic hardware configuration notwithstanding. All the software available in Windows is also available in Lubuntu software center including Arduino,

Codeblocks, Python, etc… All the apps in Lubuntu Software Center are good – it gave me a hard time deciding which one to install. All the apps worked perfectly on my old laptop.

The RAM and the CPU usage is now low (RAM is down to 29%). I actually find Lubuntu to be more interactive and attractive when compared to Windows XP.

A final word of caution however; installing and getting the distro to run on your particular machine can be difficult, and, in some cases, very frustrating. But the secret is to keep on at it, browse the web, read about other's experiences, and finally fine-tune the distro to your requirements. This effort will pay-off.
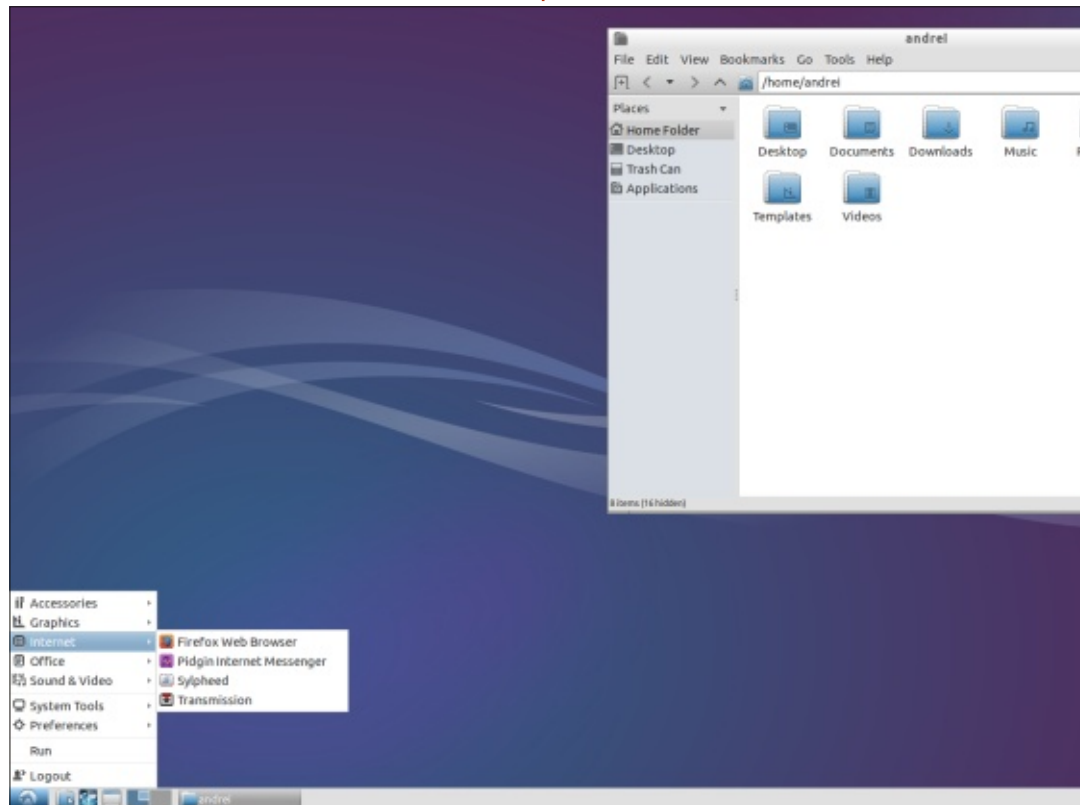
LETTERS

MORE ROBOLINUX

In light of our experience with the Robolinux Stealth VM Software, we were dismayed by the flame of Robolinux and John Martinson in FCM#86. We were excited by the possibility mentioned in issue 85 that we could finally get free of Windows after years of staying with Windows just to have Quicken, Dreamweaver, and SketchUp. So, we downloaded the software, made our small donation, and did the changeover.

Now, we're running Linux Mint. If we need to do something with Quicken, Dreamweaver, or SketchUp, we run them on a Windows 7 virtual machine that's running under Linux. The Linux and Windows applications can access the same data files, and we can copy something we've selected in the virtual instance of Windows and paste it into something we're running under Linux, or vice-versa. Short of having fully-capable open source substitutes for the Windows apps, this is just as sweet as it can get for us. The virtual Windows looks and feels like the original, accepts updates, loads quickly, and correctly runs all of the applications we have accumulated over the years.

Getting this running wasn't as easy as we'd hoped. The whole Windows installation, with all installed applications, goes through two transformations along the way. First, it is copied into a VHDX (Hyper-V Virtual Hard Disk) file (ours was 115 GB). Then, the VHDX file is converted to a VDI (Virtual Disk Image) file, which can be installed in the Oracle VM VirtualBox application running under Linux. There's an extension pack to be installed in the VirtualBox (so that it can deal with USB2 and some other things), and there's a set of Guest Additions that need to be installed once the virtual instance of Windows is running in VirtualBox (to allow the Linux /Documents directory to be shared as a drive in Windows).

We got seriously hung up on converting to the VDI file (the installer refused to accept our sudo password) and installing the Guest Additions (virtual Windows couldn't seem to find them). Through a long series of emails, the Robolinux support staff patiently nursed us through the process. Even though our particular installation had issues they had not previously encountered in more than a thousand successful installations, they never resorted to that old favorite of support staffs: blaming some other organization's product. When we reported what was happening at our end, a helpful response came back, and the responses were fresh, on-target thoughts about what to do next. It was the best software support we've ever experienced.

In fairness to the FCM#86 letter writer, we didn't try the Robolinux distribution of Linux, so we have nothing to share about that, either good or bad. And we weren't running a dual-boot machine; we were working with a Windows PC and a new Linux machine.

Jim & Celeste Parsons

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.

ENCFS GUI

I just read FC87 and want to comment on the security article. For ENCFS, there is a GUI front-end that makes it more user friendly. Just install Cryptkeeper, it is in the Kubuntu repositories. You can see how it works on YouTube:
http://www.youtube.com/watch?v=GlIfTBKK6EI

Marc Bohets

UBUNTU WOES

Two things caught my eye in FCM#87. First the article about installing Ubuntu on old computers. It mentioned the Dell Dimension 3100, which is exactly what I'm typing this on! I bought this machine nearly 10 years ago and it came with a Pentium 4, 160GB HDD, 512MB of RAM, and ran XP. Once I discovered Ubuntu (8.10 IIRC), I never looked back, but I've noticed that, as Ubuntu has got bigger, it has also got slower, at least on my machine. I upgrade religiously and noticed that each new version was slightly slower than the previous one. With 14.04,

it got so bad that I wiped the disk (after having done a full backup) and tried Lubuntu. This ran noticeably faster, but I just couldn't live with the looks (vain, I know!) and so I put Ubuntu back on and lived with it.

I should add that I have upgraded the RAM to 2GB, which is the maximum the motherboard will take, and have also upgraded the hard disk to a 2TB, 7,200rpm disk. Fortunately, I have been saving up for a brand new, high spec computer, and I hope to have that in September. I really look forward to seeing how Ubuntu will work on a core i7 with 16GB of RAM and a 2TB hard disk!

Also, I sympathize with Jan Mussche, who says that all the various distros make Linux weaker and that they should all combine to make a better distro. Unfortunately, that's the price of freedom! Being free allows us to alter things, for better or for worse. I very much agree with his comments about the command-line. I try to avoid the command-line wherever possible. Also, I rather like the idea of having an OS that very few people have as opposed to one that everyone and

his dog has. It makes you feel exclusive!

Chris Burmajster

Jan is 100% correct in the article about Linux taking over the desktop and laptop world. Everyone from all the different distros wouldn't have to come to an agreement, just some of them could form a group and follow Jan's suggestion. It would be revolutionary and like a breath of fresh air to the computing community. A lot of people don't even know what an operating system is. If you ask them what their OS is they are liable to say anything. They just want something to use to do whatever they do with a computer. They want it to just work. I know that for things to 'just work' involves a lot of variables, but I believe that it could be done. I am just a user myself that started with Windows 3.0 and Lotus 123 and got the 123 macro bug when in my late 40s – back in the early 1990s. After using XP for a while, I started dipping into the Linux pool in 2007. I bought a linux distro (can't remember the name) but it was not

that good or stable. I have used many different distros since and have settled on a couple that just keep working for me. Some might say that Canonical and Red Hat have done that but obviously not for the masses.

Harold

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 Any suggestions as to which type of server I should run to stream video and audio media?

A (Thanks to *rubylaser* in the Ubuntu Forums) I'd suggest Plex Media Server. It's super easy to set up, and has Windows, Ubuntu and IOS clients for streaming playback (even remotely, if you port forward port 32400).

Q Is there a way to remove the login drum-beat sound in Ubuntu 14.04?

A (Thanks to *Cliff_Simonds* in the Ubuntu Forums) I use Ubuntu Tweak: go to the tweaks tab ->STARTUP: login settings -> down to "play login sound" on-off.

Q How can I scan documents into searchable PDFs?

A (Thanks to *freewarelover* in the Ubuntu Forums) Install gscan2pdf. It has an Optical Character Recognition (OCR)

function.

Q I just installed Ubuntu as a remote media player. How can I control it from my Windows 7 computer?

A (Thanks to *HeroHog* in the Ubuntu Forums) Use X11VNC on Ubuntu, and TightVNC in Windows.

Q I'm thinking of getting a laptop. How can I find out which ones will work well with Ubuntu?

A There are two message threads in the Ubuntu Forums which might help, and you can get to them from Google. One is the "laptop compatibility list," and the other is the "laptop incompatibility list" I don't trust the latter; some people think the microphone doesn't work when it's just muted...

If you are prepared to pay a little more, there are vendors which sell laptops with Ubuntu pre-

installed. My preference is to use unetbootin to put the latest Ubuntu on a flash drive, and boot from it at the dealer's. See if sound works, the webcam (install Cheese), wireless, etc. (If you can see wireless networks, you don't actually need to connect to one.)

Q I'm using Ubuntu 13.10 and ...

A Before you can fix your problem, you need to update your system to a version of Ubuntu which is currently supported. As of right now, that means either 12.04 or 14.04 for the desktop versions.

Q To compile my first program, I typed 'gcc -o test main.c'. However when I run the program in the terminal, I don't see anything.

A (Thanks to *trent.josephsen* in the Ubuntu Forums) Test is a shell command. To run your compiled program, use this command:

```
./test
```

TOP QUESTIONS AT ASKUBUNTU

* Ubuntu on low performance laptop
http://goo.gl/GlixGy

* Why is this password not good enough?
http://goo.gl/gDez1N

* Why does aptitude insist on installing packages?
http://goo.gl/H2OPgq

* how to know if an authenticate box is real or fake?
http://goo.gl/IgWK0q

* Watching Netflix on Ubuntu 14.04
http://goo.gl/GqCp9K

* Does spyware exist for Ubuntu?
http://goo.gl/u0kJis

Q&A

* What version of Python do I have?
http://goo.gl/A5LEQt

* Skype can't connect
http://goo.gl/6CiEOG

* How to update LibreOffice from 4.2 to 4.3 in Ubuntu 14.04.1 LTS?
http://goo.gl/eUGfXv

TIPS AND TECHNIQUES

Server: Hoist by my own Petard

Many times, a person has asked a question about Ubuntu Server, and I have responded, "use a GUI, where all the familiar tools are available. Try Xubuntu, Lubuntu, Mint with Mate, etc." Then it happened: I needed to re-purpose an older computer as a special-purpose file server, with the maximum possible performance.

I looked at options. Windows 8.1 Desktop is fine as a limited-purpose file server, but it probably won't deliver "the maximum possible performance." FreeNAS has a great reputation, but it's a memory hog. OK, Ubuntu Server it is -- with no GUI.

My starting point was a core duo computer with no hard drive. I added a hot-swap drive bay so I can change the shared drive when it fills up, and began with a 2 TB drive as the shared file space. The computer is so old that it won't boot from USB, so I installed a small hard drive to hold the OS.

Installing Ubuntu Server isn't much of a challenge, but there are a few tricks. I found that my life was a lot easier if I had Internet access during the installation, so the installed system would assume Internet access was available. During installation, you are asked what applications you want. First and foremost, I wanted Samba file server. When I also selected OpenSSH Server, I could control the server from another computer -- from the very first time it booted.

I wanted to run a "headless" server with no monitor and no keyboard, but the Dell BIOS complained, "keyboard failure, press F1 to continue," and would not boot. OK, I had to leave the keyboard attached.

To control the computer from Linux, I installed PuTTY from the repos on my normal desktop computer. To control it from Windows, I went to putty.org and downloaded the client program. One small, but important point: I have told our router that when the server's MAC address appears, it should get the IP address 192.168.1.70, so the file server has a static IP address on our LAN. Putty is a lot easier to use when it is connecting to a static IP address.

So now I'm at the command line, and need to master a few specifics.

The first question is, how to automatically mount a second drive, which is not what I booted from. This helped:
http://www.ghacks.net/2009/09/10/add-a-second-drive-to-your-ubuntu-server

However, the first comment suggested using UUID, which took a little bit more research. The key command: sudo blkid

Next, I needed to understand how to set up a shared folder using the command line. This helped:
https://help.ubuntu.com/10.04/serverguide/samba-fileserver.html

Even so, it took some experimenting to get the shared folder on the second drive, and make it mountable and writable from a Windows client computer. The key command:

```
sudo chown nobody.nogroup
foldername
```

So, three technical issues to master, and all of them have good online tutorials. Because I already understood the terminology, it was much easier than I expected.

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 Iain Mckeand

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.

On Thursday, March 13, 2014, I had a scary experience. I was using my computer remotely, using teamviewer, and opened a link in an advertisement to the website www.optionrepublic.nl

This site was opening very slowly. I used Chromium and there was a message in the lower left corner saying it was waiting for some other website. I was at work, so I did not have the time to wait and continued working, leaving the chromium window open and minimizing the teamviewer window, so I could not see what happened at home. After some time I decided to look if the page has loaded and restored the teamviewer window. What I saw scared the hell out of me. The mouse was moving, opening the menu, starting the terminal.

Some wget ... command was copied and executed and the resulting ...linux.tar.gz file expanded. When nano was started to edit some (system) file I unfroze. I did not want to wait to see if the intruder also found a way around

the system password and simply shut the computer down.

I do not think the intruder hacked his way into my computer through teamviewer.

I think the site I opened was a malware site, downloading and opening a teamviewer-like program to take over my computer. If I did not see it happening, he would have started a trojan horse, hoping to catch my password and install something at system level. I changed my teamviewer password as well as several others, using my not-infected laptop and reinstalled Ubuntu from scratch on my home computer.

This experience proves we may not assume that Linux is safe to use and cannot be infected. Through a bad website all kinds of software can be loaded temporarily and used to infect your computer.

So, in conclusion, the best advice is: if it takes long to load a

website: close the window, something bad is creeping in.

Michael Boelen: Malware on websites is definitely a common threat. The risk of getting infected via this method can be lowered by avoiding suspicious websites. Another proper way is to keep your browser fully updated at all cost. Each flaw in the browser will sooner of later be misused, directly via the website you are browsing or via a third-party website (e.g. advertising). In this case, Theo might be right and the website might be slow due to the malicious code freezing up the browser. Generally this is the cause due to the code trying to fit in "wrong data" in different memory locations, with the hope to load a malicious payload. On the other hand, a website might be simply slow because of the server being busy, or dealing with way too many clients and waiting for resources to free up. Advice I would give the readers is to use an up-to-date browser. Use addons like NoScript to prevent any javascript or similar from running (unless you trust it).

Also scan your system on a regular basis with tools like ClamAV, Lynis, Rootkit Hunter, or any commercial anti-malware scanners.

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.

Who says that video gaming is not a spectator sport? This year, while the 2014 World Cup in Brazil was wrapping up and Germany was crowning themselves victors, the largest video game championship was just getting underway. This event is called The International 2014, and it's a yearly event that draws players and viewers, from all over the world. This year's total prizes topped $10 million dollars! Yes, you read that right, there were over $10,000,000 in prizes, with the first place team getting just over $5 million. Having broken the record of largest prize-pool in eSports history in 2014, The International is undeniably the single biggest event in eSports today. So what is The International, and why is it such a big deal? Let's find out.

The International is an annual eSports video game championship event hosted by Valve (the company behind Steam) in which teams of five players each compete by playing the game Dota 2, and the top 14 places all receive some sort of prize. The International

2014 marks the fourth consecutive year in which this competition takes place. Its popularity growth is evidenced not only by its ever growing prize-pool money, but also its ever larger viewer base. This year, the Semi-Finals as well as the Finals were even broadcast on ESPN3, which is evidence that it is a profitable event for broadcasting companies even though most of its viewer base comes from online Internet live streams.

The International 2014 was hosted by Valve Corporation, which is the company responsible for the popular Steam video game engine. This year, The International took place from July 8 through July 21 at the KeyArena Center in Seattle, Washington. There were 11 teams from all over the world directly invited to this year's tournament, in addition to 41 teams who participated in the qualifiers. Much like other big sport events (e.g., FIFA World Cup), the qualifiers are

split into four groups. The four qualifiers groups are Europe, the Americas (North, Central & South), China, and South-East Asia. The four winners of the groups are then invited directly to Phase Two with the 11 invited teams while the four runner-ups go on to Phase One to fight it out and move on to Phase two. This makes it a total of 19 teams participating in the actual tournament.

According to Valve, there were over 20 million viewers streaming the event this year. According to CBCNews, the event had more Internet traffic than Facebook. You could stream the event from the Dota 2 The International website, from within the game Dota 2, as well as other sources like ESPN, twitch.tv and Xbox. It is still possible to watch any of the matches from the event; of course it won't be live because the event has already ended. Whichever option you choose to watch the event, it shares similar qualities as other sporting events but with a modern twist. For example you have a choice of commentators.

UBUNTU GAMES
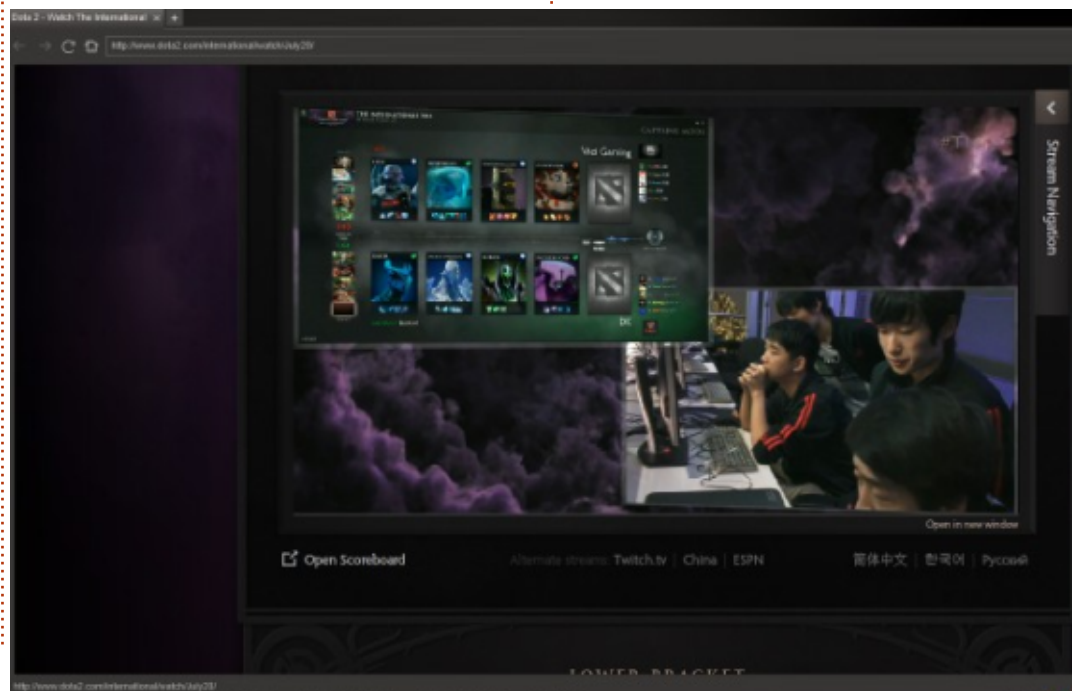
From within the game, you're able to choose what commentators to listen to (they're available in a variety of languages). There's also the choice of which camera(s) to watch. My personal preference was to have the commentators and the camera angles matching so that the commentary is based on what I'm watching.

Valve was able to get some of the money for prizes from sales of The Compendium, which is a digital interactive book intended for fans

to participate in the event as well as to raise funds for the prize-pool. The basic Compendium was being sold for $10 out of which $2.50 went directly to the prize-pool. Fans who purchased the Compendium would benefit in the form of getting a few gifts, like various Dota 2 loading screens, in-game landscape effects (fog, snow, etc.), alternative voice-overs, and a boost in points earned in Dota 2 during the duration of the event. If you completed certain challenges from the Compendium, you would

be rewarded with more gifts, like a higher point boost, more loading screens, etc. Some of the challenges were activities like watching 10 qualifier matches, watching 10 playoff matches, etc. You also had the choice of selecting your picks for winners, selecting which Dota 2 characters would be chosen the most, the least, or would have the most kills, earliest kill, etc. Also, there were challenges like the Daily Hero Challenge in which you were given 10 Dota 2 heroes that you had the

option to play with in your regular Dota 2 game, and, if you won with all 10 of these heroes, you would be rewarded with 100 Compendium points which increases your Compendium gifts.

The first place team was Newbee from China, who battled it out in the Main Event against second place ViCi Gaming in a best out of five Grand Finals decided in four matches. It is a rather comforting thought knowing that the largest event in eSports history is based on a video game that is currently available for Ubuntu Linux through the Steam game engine. This is certainly a great sign of things to come for Linux gaming in the future. Lookout for next year's The International tournament which is sure to be an even greater success than this year's event.
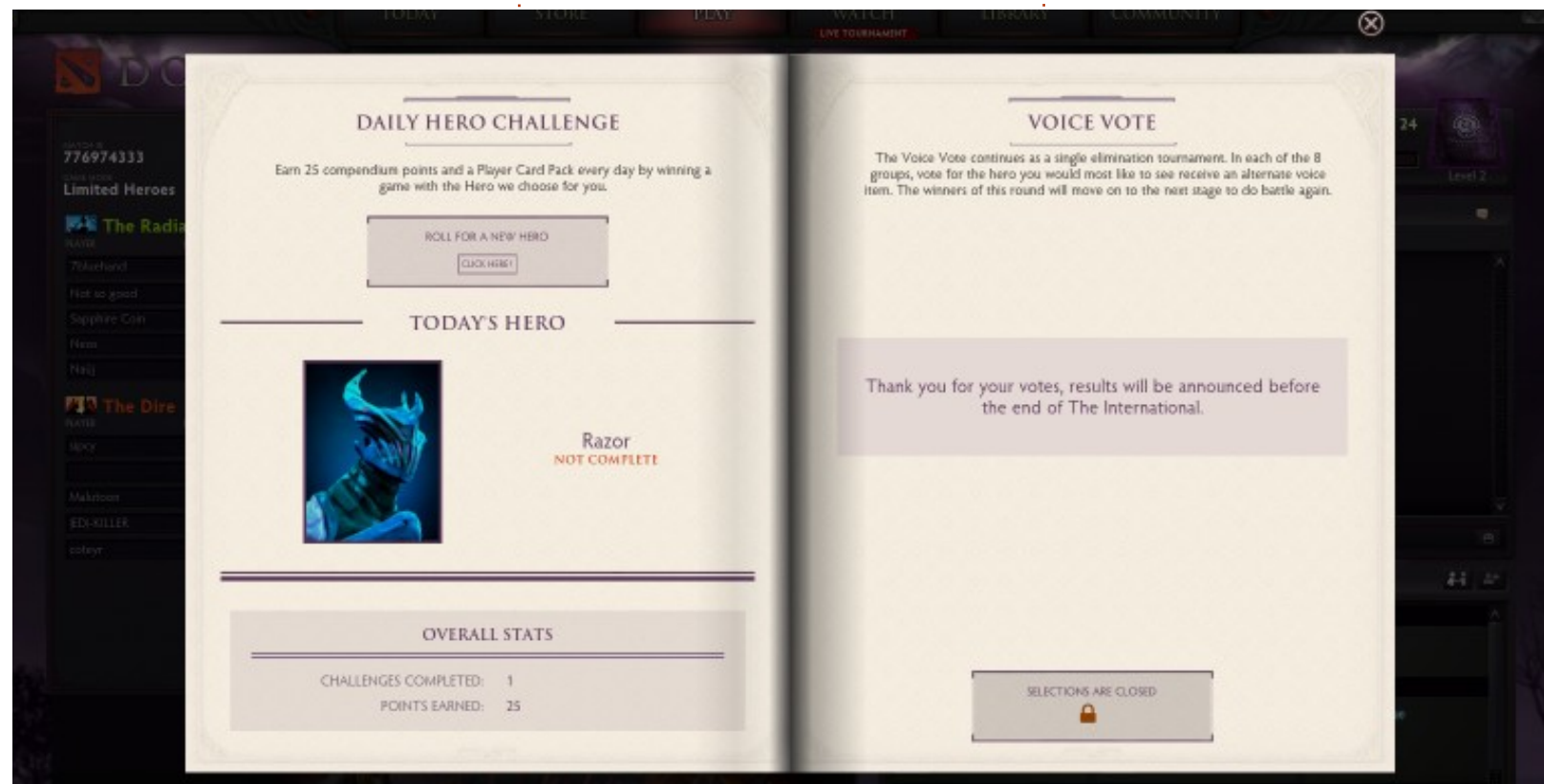
Oscar graduated from CSUN, is a Music Director/Teacher, beta tester, Wikipedia editor, and Ubuntu Forums contributor. You can contact him via: www.gplus.to/7bluehand or email: www.7bluehand@gmail.com

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.
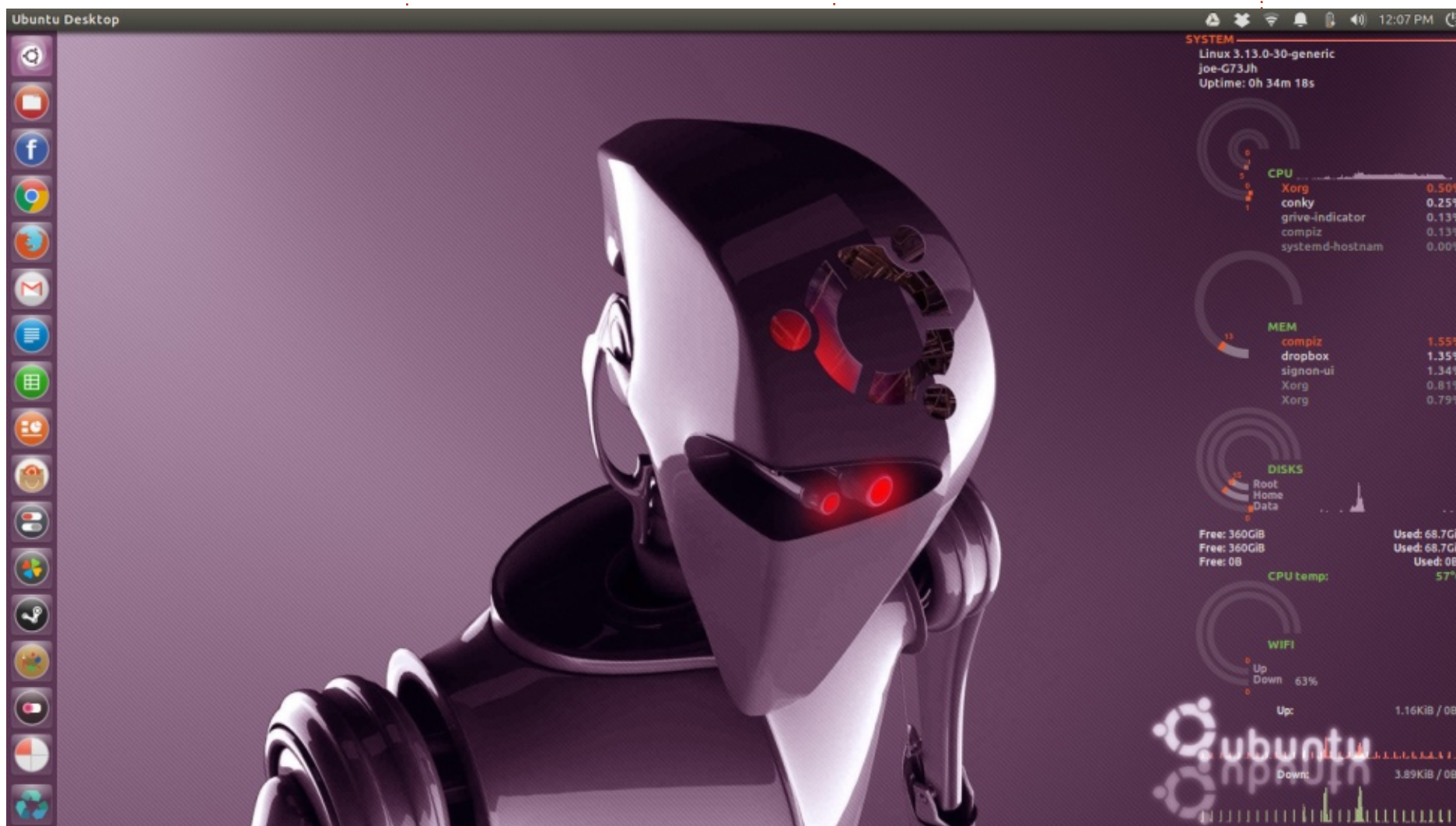
My computer is a g73jh Asus laptop that I primarily use for gaming. It has an i7 processor and 8GB of RAM. I wanted my desktop to appear futuristic because I believe the world of gaming has a lot in store for Linux. I am using Ubuntu 14.04 with the unity desktop and the numix circle theme for a more futuristic look. I have conky configured to measure all of my computer's vitals during a gaming session.

Joe Covell

MY DESKTOP

This is kubuntu 14.04, so I use the KDE desktop:
Boot animation: CodeKde
theme: Invisible
Window decorations: Air-black
Cursor theme: Pulse-Glass
Desktop Theme: Invisible

Desktop picture: abstraction_vector_girl_headphones_318_3840x2400
Desktop widgets: hardware information, memory(swap-RAM), CPU frequency

My PC is notebook HP g62:
CPU: AMD Turion 2 P540 (Dual-Core)
GPU: amd/ati radeon HD 4250 RS880 (open driver)
ram: 4 GB DDRIII
HD: Sata 500 GB

Other OS: manjaro 0.8.10(arch) and elementary OS Luna(ubuntu)

Daniele

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#89
Deadline:
Sunday 14th Sept. 2014.
Release:
Friday 26th Sept. 2014.

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.

Google Play - You can now read Full Circle on Google Play/Books. Either search for 'full circle magazine' or click this link: https://play.google.com/store/books/author?id=Ronnie+Tucker