BUILD A PI-POWERED BIGTRAK

14-pages of practical Raspberry Pi guides and features

www.linuxuser.co.uk

LinuxUser & Developer™

FIND ALL THE TUTORIAL FILES YOU NEED
www.linuxuser.co.uk/tutorial-files

KANO

"Simple as Lego, powered by Pi"

Interview full guide & review

Master the command line

Terminal tips and tricks to try today

wattOS R8 reviewed

Is this the greenest distro ever made?

Program Android apps with Java

Make amazing apps for Android

SWITCH TO LINUX

XP is EOL, Windows 8 is DOA. Make the switch today with our **expert 8-page guide**

ALSO INSIDE
» Duo Security reviewed
» Manage invoices with jBilling
» Two-step authentication with Twilio

Synology DS414j

Dual core and four drive bays: Is this the ultimate NAS?

IMAGINE PUBLISHING
DigitalEdition
GreatDigitalMags.com
ISSUE 141

CYBERHOSTPRO
www.cyberhostpro.com

RESELLER HOSTING

START YOUR WEB HOSTING BUSINESS TODAY FROM JUST £9.99 MONTH

- Host Unlimited domain name
- 100% Branded to your business
- Easy to use web based control panel
- Wordpress installer
- Plus much more

WEB HOSTING FROM £2.49 month

- Up to 10GB Space
- Premium Email Accounts
- Host up to 30 websites
- Web Based Control Panel
- Windows 2012 R2 or Cloud Linux 6
- Instant activation

CLOUD VPS SERVERS FROM £15 month

- Build a server to your exact requirements
- Scale up to 32GB Ram
- Fail safe hardware
- One click backup & restore
- Catalogue of Operating systems
- Plesk or CPanel also available
- Instant activation

DEDICATED SERVERS FROM £99 month

- Latest Dell Poweredge servers
- Multiple Intel Processors
- Up to 8TB disk space
- 10TB monthly bandwidth
- Windows or Linux OS available
- Plesk or CPanel also available
- 1Gbit port

Hosting Heroes

Our heroes are available to support you 24 hours a day, 7 days a week for any queries or support you may have with your hosting services with us.

No Contracts

All our products are contract free. You can cancel at any time by giving us notice before your next billing due date.

99.9% uptime guarantee

We have a 99.9% uptime guarantee to make sure your website is always online.

Use coupon code **WEBDESIGNERMAG** to receive 10% off todays order.

08455279345

Welcome

to issue 141 of Linux User & Developer

Your team of Linux experts...

Rob Zwetsloot studied aerospace engineering, using Python to model complex simulations. This month Rob threw down his archaic shades and espoused the virtues of Linux to the Windows world (pages 24-31). He got frustrated explaining why there's no Clippy in Linux and has since disappeared into his testing lab.

Richard Smedley started using computers long before WYSIWYG and still maintains that the command line, and Emacs, is the most productive working environment. In this issue Richard shares with us his terminal tricks for making scheduled file backups and general system maintenance a cinch (pages 34-37).

Tam Hanna has been in the IT business since the days of the Palm IIIc. Serving as a journalist, tutor, speaker and author of scientific books, he has seen every aspect of the mobile market more than once. This month Tam embarks on the first of a three-part series on building Android apps with Eclipse and Java (pages 50-53).

Sean Tracey is a creative technologist at a leading digital agency on the south coast. He spends a lot of his time living inside of Node.js, Python and Arduino. This time Sean shows us how to set up two-step authentication in web services using Twilio's SMS API to automatically text verification codes (pages 46-49).

Jon Masters is a Linux kernel hacker who has been working on Linux for some 18 years, since he first attended university at the age of 13. Jon lives in Cambridge, Massachusetts, and works for a large enterprise Linux vendor. You can find his brilliant Kernel Column on pages 22-23 this month.

Gareth Halfacree is our resident news reporter and brings us the latest developments from all over the open source world, starting on page 14. Find out what he thought of Duo Security's smartphone-based authentication system and Synology's DS414j on pages 78-81.

This issue

- Spring clean with the CLI
- Code your own Android app
- Help a friend switch over to Linux
- Drive a Bigtrak with a PS3 controller

Welcome to the latest edition of Linux User & Developer, the UK and America's favourite Linux and open source magazine.

Kano, the Kickstarted computer kit pegged for release in just a few weeks, may well be the innovation that drives Raspberry Pi further into the public imagination and captures the attention of a new generation of coders. Aiming to remove all barriers to entry, this Pi-powered computer comes with two project books and a Kano OS that inject even more fun into programming. You can find out all about it starting on page 6, where we interview founder Alex Klein and review the built-in Kano OS.

If you know anyone still hanging on to Windows XP, now is definitely the time to show them the light of Linux. Turn to page 24 and you'll find our complete switcher's guide to help.

Elsewhere in this issue you'll find plenty of projects to sink your teeth into; from emulating Ubuntu Touch and coding Android apps to sensing motion with the Raspberry Pi and using it to build a tweeting flood warning system, we've got you covered. Enterprise readers will want to check our guide to streamlining invoicing with jBilling too.

Finally, I'd just like to say that I'm thrilled to join you all here at **Linux User & Developer**, and wish Russell Barnes all the best (and all the best robots!) for the future.

Gavin Thomas, Deputy Editor

Get in touch with the team:

linuxuser@imagine-publishing.co.uk

Visit us online for more news, opinion, tutorials and reviews:

www.linuxuser.co.uk

LinuxUser & Developer

Contents

Subscribe & save!

32 Save up to 50% on the shop price. US customers can subscribe via page 84

06 Kano: Simple as Lego, powered by Pi

The new computer kit that makes Pi coding easier than ever

OpenSource

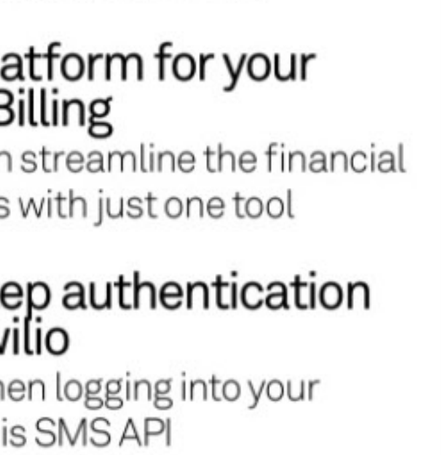
Features

Tutorials

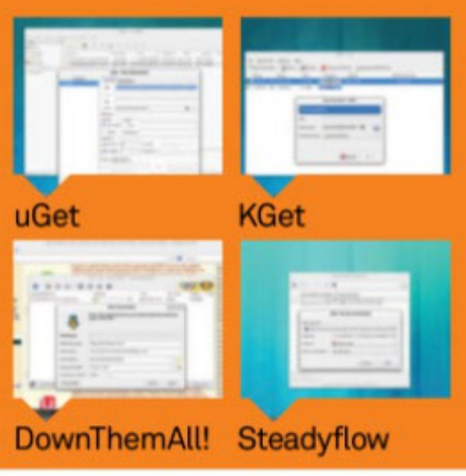
On your free disc

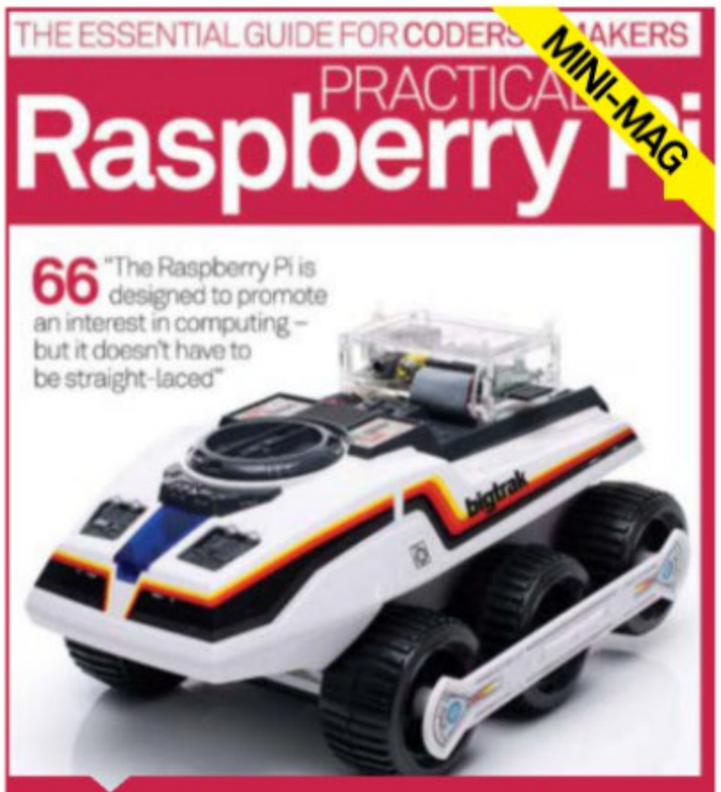
96 Cover disc
Four of the latest distros for you to try out on this issue's DVD!

Linux Mint 17
Fedora 20
Mageia 4
openSUSE 13.1



Reviews





Join us online for more Linux news, opinion and reviews www.**linuxuser**.co.uk

FRESH. FAST. FREE.

LIBRE.

The Document Foundation
proudly presents:

LibreOffice

the world's most advanced free software office suite

www.libreoffice.org

KANO

Simple as Lego, powered by Pi

For some time now, there has been much talk about the Raspberry Pi revolutionising the teaching of computing in schools. Linux User & Developer has devoted much space and attention to the growing number of Jamborees and the increasing attention teachers are giving to the small, £25 bare-bones machine. It is, say advocates, the perfect way to introduce children to the world of computing, allowing them to see and actually interact with the innards of the machines they are using. It is, they add, a great platform for programming and for creating all manner of electronic wonders.

But for former journalist Alex Klein, it doesn't quite go far enough and – simple as many believe it is to use – he believes the Raspberry Pi in its vanilla state is still too confusing for some. He points to the *Raspberry Pi For Dummies* guide which, at 400 pages, he feels is far too long and impenetrable. This is the reason why he began a Kickstarter project called Kano: in order to produce a user-friendly computer and coding kit, asking for $100,000. By the time the project was successfully funded on December 19, 2013, 13,387 backers had pledged $1,522,160.

Far from setting itself up to be in competition with the Raspberry Pi, the Kano is actually powered by it. Instead of giving people a simple board, Kano wraps the Pi in a case, adds all of the necessary cables and bundles in a colourful keyboard. It also runs a version of Linux that has been designed by the Kano team and places an emphasis on programming. Rather than ask people to fiddle around with the Pi until they figure out how to use it, Kano expects users to turn it on, instantly get going and figure out the deeper stuff later on. And, unsurprisingly, the idea has really taken off.

What's in the box?

1. The **books** you get come with everything you need to set up your Raspberry Pi and then learn how to use it and its teaching tools.
2. The custom wireless **keyboard** and trackpad provided is called a magic wand in the literature. It connects wirelessly to the Kano and Pi.
3. Included are three sets of **cables**: HDMI for video and audio, microUSB for power and an ethernet cable as an alternative to the wireless dongle.
4. The Kano OS is pre-installed on the custom 8GB **SD card**, which includes all the apps and games that make up the Kano learning program.
5. A low-profile **wireless dongle** allows you to put the Pi anywhere there's a screen; there's even a section in the manual for setting up Wi-Fi.
6. Kano calls this the brain of the kit, but we all know it as the **Raspberry Pi**. It's as important as the Kano OS to help teach computing.
7. The **main case** for the Raspberry Pi is customisable and easy to assemble, keeping it safe and making it look like a real computer.
8. The **covers** for the case allow kids to personalise the computer they just made with a variety of different colours.
9. If you're not plugging the Pi into a normal TV, you may not get any sound. The **speaker** case mod allows you to get around that.
10. A good **power supply** is important to properly power the Raspberry Pi. The Kano kit comes with one specifically tailored for the job.

How the package stacks up

The Kano kit is an integrated experience. Core to Kano is the Raspberry Pi Model B. But there are also two project books that are 150 pages long, with 12 levels of various games, apps and coding adventures.

"We tell it all like a story, which is one of the reasons we did them in book form," says Alex. "There has also been interest in the SD card that comes with Kano OS so we're selling that independently."

The DIY speaker, which is quite disruptive, is the coolest audio add-on you could get for the Pi as a separate product. The Kano keyboard is designed to be as suitable for PC gaming as it is for working in a text editor. "A lot of people really like that it has an integrated mouse and two touch click buttons," says Alex.

"The key product is the Kano Computer Kit, as we call it – the computer that you build and code yourself, rather than just consume."

Kano is also looking towards the future: "We want to start rolling out new expansions, a bigger display, build the battery, build the server, build the phone booth, build the robot," says Alex. "We're really focused on creating a 'wow' experience for today's 13,387 people that originally backed us."

KANO

Kano's popularity demonstrates just how many people share Alex's vision and belief – but was he surprised?

"To an extent," he says. "When we launched Kano, we figured the kits were costing us around $90 to make, so that's why we came up with the figure of $100,000. We thought that we'd make 1,000 kits and then keep testing, keep iterating, keep going into schools, keep talking to artists and engineers and parents and make it better. But when we pushed the button, something crazy happened and the idea just sort of took off. We raised $100,000 in 16 hours."

The idea for the Kano had actually been formed in January 2013 with the original challenge for the project coming from Alex's seven-year-old cousin, Micah. The youngster had said he wanted to build a computer that was as simple and as fun as Lego and Alex, who was doing some graduate work in Cambridge and was in contact with Eben Upton from the Raspberry Pi Foundation, decided to introduce his cousin to the wonders of the Raspberry Pi.

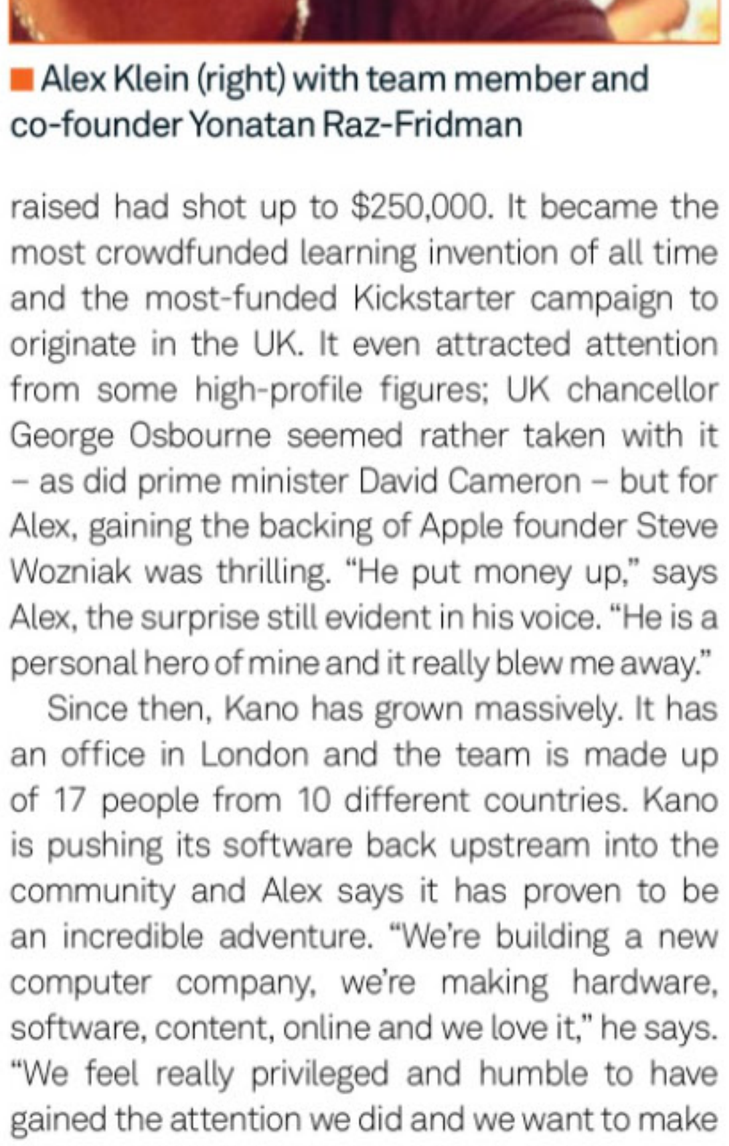
The power to program and play is now in the hands of the children using Kano

"But it was at a time when there wasn't too much available for it and there wasn't a lot of software," Alex says. "I was incredibly excited about the Pi and I wanted Micah to take to it, but it was essentially incomprehensible to him. I realised that kids today have grown up in a world of hermetically sealed technology, so when you even try and expose them to the shell or open source libraries, they don't quite get it."

Alex decided he would take the Raspberry Pi, power it up with intuitive peripherals and software, produce some projects in the guise of games based around coding and make it accessible for the mainstream. He realised that not everybody was into the Pi-maker and hacker community like himself, so he felt this new approach was necessary.

"That's how Kano came about. I needed it to be about playing, making, producing music, creating a Pong game or building a massive structure in Minecraft," he said. "It had to be familiar. So we worked tirelessly on the design. We worked with Eben and we went to hundreds of schools, tested early versions of the kit, kept iterating and improving. We created these little DIY speakers that drew power from the Pi's GPIO ports and we created a cool modular side-clipping case that allowed you to extend your Pi in numerable ways while still clipping into a basic framework. All along, we had in mind that the user shouldn't get confused; they could basically still always look back and make sure that everything was attached appropriately."

In the process, Kano has proved to be a record-breaker. Building a computer from the PCB up, making it intuitive and basing it on storytelling resonated with people. American-born Alex flew from London to the USA to show his friends and family what he had been working on. By the time the plane landed, the amount raised had shot up to $250,000. It became the most crowdfunded learning invention of all time and the most-funded Kickstarter campaign to originate in the UK. It even attracted attention from some high-profile figures; UK chancellor George Osbourne seemed rather taken with it – as did prime minister David Cameron – but for Alex, gaining the backing of Apple founder Steve Wozniak was thrilling. "He put money up," says Alex, the surprise still evident in his voice. "He is a personal hero of mine and it really blew me away."



Alex Klein (right) with team member and co-founder Yonatan Raz-Fridman

Since then, Kano has grown massively. It has an office in London and the team is made up of 17 people from 10 different countries. Kano is pushing its software back upstream into the community and Alex says it has proven to be an incredible adventure. "We're building a new computer company, we're making hardware, software, content, online and we love it," he says. "We feel really privileged and humble to have gained the attention we did and we want to make good on it by shipping something amazing."

And ship they will. Kano told backers from the start that the computer would ship in the last week of July, so people will soon be getting hold of their bright orange boxes and delivering their own verdict on whether the project has

> "Kano has become the most crowdfunded learning invention of all time"

Micah, the inspiration for the Kano, has a play with an early prototype and enjoys some books

been a success. There is always a danger that the project will fall short – there are hundreds, if not thousands of Android-based Ouya consoles gathering dust and being flogged on eBay, for example – but it seems unlikely Kano will go the same way. Alex says Kano will not stand still and his team is already looking to further and expand on what has already been achieved.

One of the key achievements has been Kano OS, a fully featured operating system for Raspberry Pi that is based on Debian Linux. The website says it has a simple interface, allows for seamless setup, automagic Wi-Fi and Updater and coding projects. Plus, it can be downloaded and put on an existing Pi using the Kano Burner, so everyone can try it. There was never any question of basing the OS on anything but Linux.

"I'm a Linux hobbyist and amateur," says Alex. "It's not my main machine but it's something that I've always experimented with in the playground or, as they call it, the Bazaar – which people will know if they've ever read *The Cathedral And The Bazaar*, which is just a brilliant book on Linux and open source. The future, whether people accept it or not, is going to be driven by open source development. Debian on ARM, especially in the non-Western world, is becoming the platform of choice for creative, distributed ground-up development. It's not surprising. We tend to assume that the best tech products have to emerge from the hermetically sealed Jonny i-run Apple lab in Silicon Valley, but the truth is we're

Kano is easy to set up and the interface is built so that it is incredibly child-friendly

Kano levels

Video A command line YouTube app that has several uses, with its main ability to do a limited search of YouTube to watch videos from. In addition, it helps teach children how to use the command line to access programs. It also has the secret purpose of making sure your internet and sound is working just fine.

Snake The classic videogame is recreated here for the command line, going through the basics of appending and using options. The game is fully editable as well, allowing users to change its colours, size and speed as they follow along with a series of easy-to-understand tutorials told through the command line itself.

Pong Kano invites kids to play one of the very first videogames in a recreation of *Pong*. This time, editing and customising takes the Scratch approach, using blocks of code to create variables, loops and functions. The tutorials take users through editing the game to their liking and improving it overall.

Minecraft Available as the full Pi version of the game on Kano – the first Raspberry Pi distro to do so – and as a special version users can edit and program. It's in the form of Scratch-like code blocks and includes tutorials that teach kids how to use the editing tools, allowing them to edit it once the tutorials are finished.

Music While the previous challenges teach the way code is laid out, the music section lets kids put it all together and actually write some code thanks to Sonic Pi. Once again, they'll be taken through the motions of creating little melodies, which involves writing loops and taking human input.

KANO

The Kano keyboard

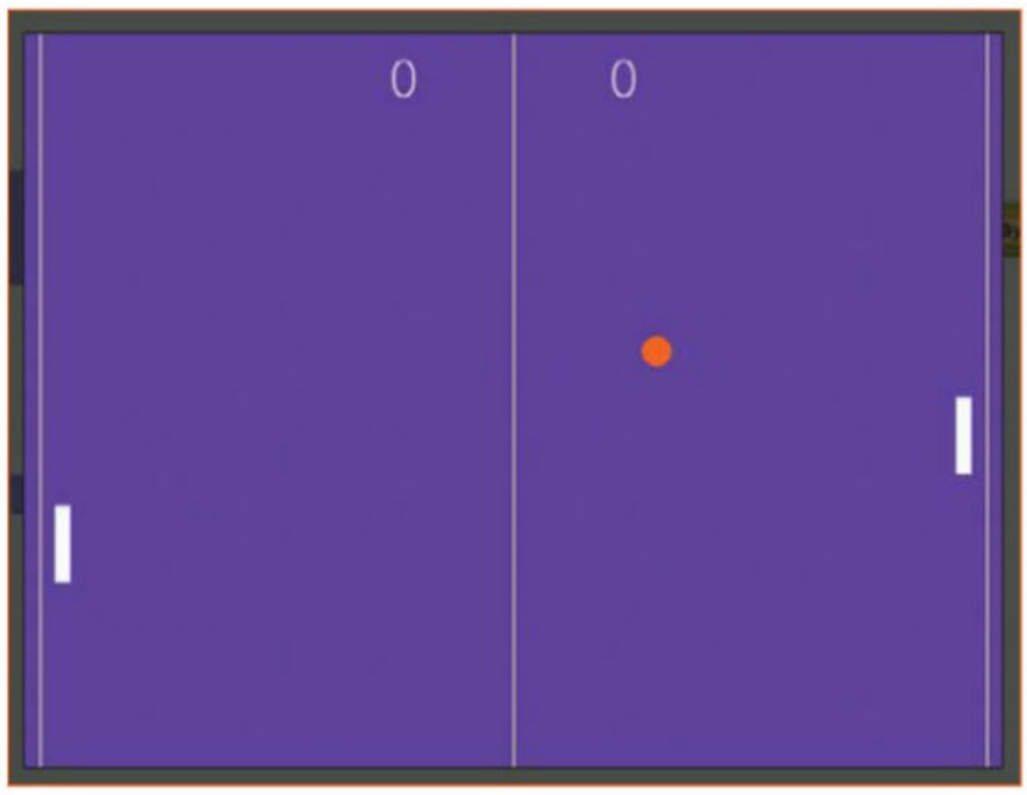
When you get down to it, the Kano kit does come with a lot of simple components you could easily pick up elsewhere – but that doesn't change the fact that it's great to have the necessary, high-quality components all included in one kit. One of the standout components is the wireless keyboard with built-in trackpad; one of the custom-made pieces that comes with the kit.

The slimline orange device fits the aesthetic of the Kano project perfectly, while still being useful and very portable. Also known as the magic wand in the manuals, it uses a dongle stored on the underside to connect to the Pi, allowing for the wireless dongle to attach to the other USB port in the process. There's also a small wired USB cable if you need to connect it physically as well.

The keyboard is quite full-featured, with the standard keys you would expect to find along with some alternate function keys. These keys include the usual media options, some extra punctuation keys and, more interestingly, some Kano-specific keys such as Save, Load, Make and Share. These keys are tied to particular functions in some of the editing tools – such as the *Minecraft* programmer – and can make using it much faster when you know how.

The trackpad on the side is quite high fidelity and perfectly placed for right-handed playing of *Minecraft*, with WASD on your left. To maximise the space the trackpad takes up, the right- and left-click buttons are separated into standard keys on the keyboard and placed on the other side of the device itself. For a lot of the Kano editing tools, a two-handed mouse approach works just fine and you'll be able to use the standard tap gesture with the trackpad to perform a left-click.

It's an excellent and well-built device perfect for someone's first steps into proper computing, especially in an age where kids are learning more and more exactly how to use computers, whether they're on a tablet or smartphone.



■ Users can play Pong on the Kano, giving a sense of its retro roots

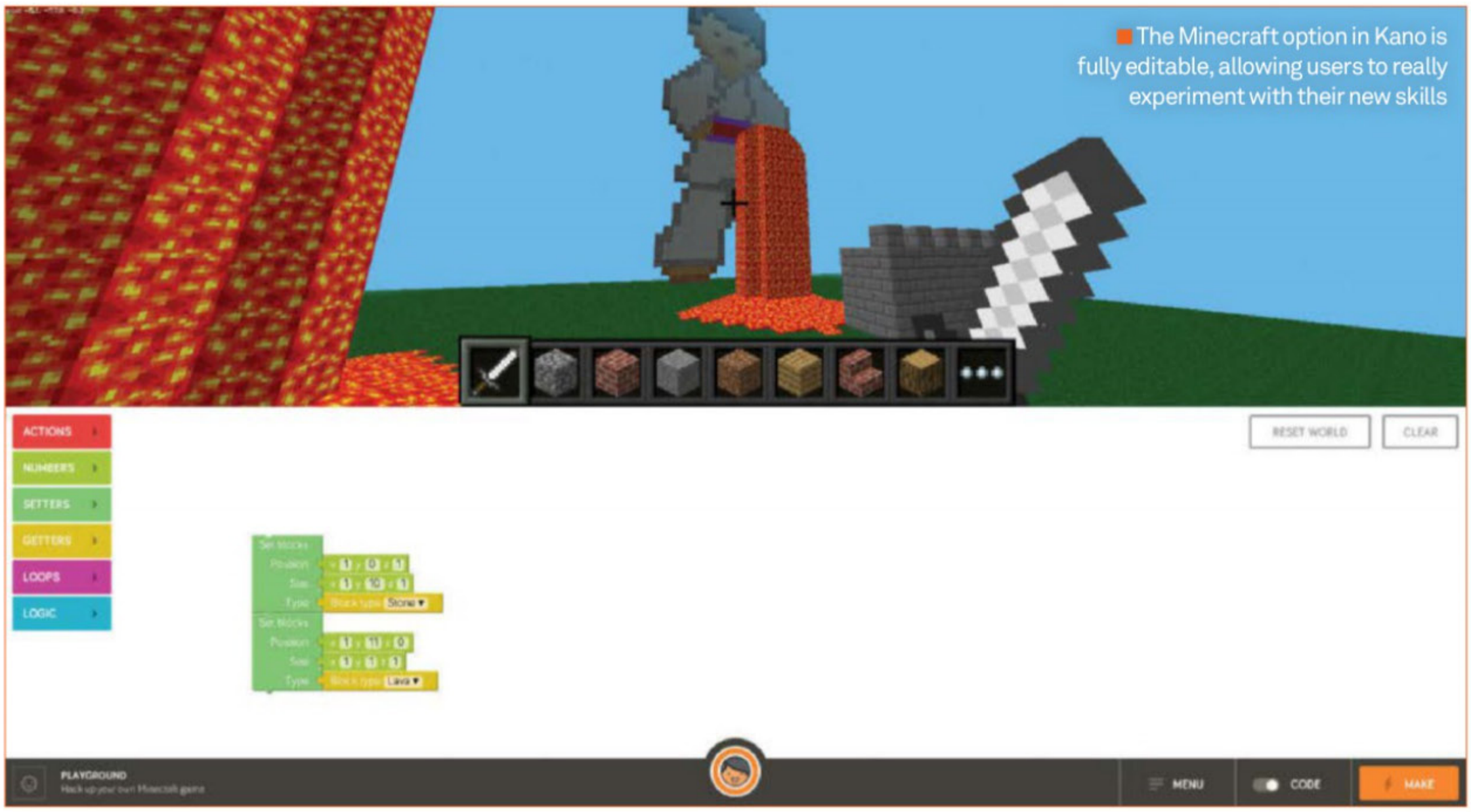
■ The Kano Keyboard has been custom-built for purpose

reaching a point where Stephen Frost [Debian's maintainer] can compete on all levels with closed and non-lever solutions."

Alex is certainly an advocate of Linux. He points out its use in sub-Saharan Africa, in East Asia and in the Indian sub-continent, and he says we're seeing ever-more younger people adopting free and open-source stacks. As well as affordability, he says, open source allows for a much more anarchic kind of marketplace of ideas. He wants Kano to be the company that takes this spirit to the mainstream, making it simple and fun for anyone anywhere to make and play with Linux.

"I was turned on to the magic of Linux at a very early age," Alex continues. "I'm young enough that I never really played around that much with DOS, so this is a type of magic that the younger generation deserves the chance to experience. What's great about the Raspberry Pi is they've created a cheap modular Linux board at a price less than a textbook. People like us, we can come in and kind of Lego-ify the experience and make it as sort of ground-up and constructionist as your first build-a-helicopter set that you did when you were seven with Lego."

But like Lego, Kano isn't a charity. It is run on a business model and it aims to make money. Saul Klein, the father of seven-year-old Micah, is a big name in the venture capital community and he helped to fund the venture in its pre-Kickstarter days. Alex found Kickstarter liberating because it enabled him to build a rapport directly with his customers, but it is still a commercial venture. Not that there is anything wrong with forming a company around Kano – the more money it attracts, the more ambitious its aims can be.



■ The Minecraft option in Kano is fully editable, allowing users to really experiment with their new skills

There is a sense that, for Alex, the fun is in building something from scratch and being able to get involved in the entire process.

"It brings a bit of humanity back to software development," he says. "I like the level Kano is at. With open source and Linux, you see something you like, whether it's raindrops rolling down a window in JavaScript or a beautifully customisable desktop manager for the Raspberry Pi, or an open source music synthesiser based on SuperCollider, and you like it and you fork it, and you get to know the creator and sometimes they're just an amateur, it's a hobby. That's the beauty of Kano: it's like you build a relationship and that's how kids are naturally. They're not precious about what they make. They make things in order to be cool. I think as we get older we start drawing lines around things and we say, 'Yeah, we're going to protect this, we're going to copyright that'. That's fine, that's part of the creative process as well, but when you're young there's a sense these days, especially with this generation, that you have an audience from the minute you go on the internet. You have a community, you're surrounded by people who are already interested in what you want to do."

He talks of 14-year-old Amy Mather who is making a name for herself in Raspberry Pi circles and her use of open source libraries to create Conway's *Game Of Life*. He discusses children from Westminster who invented the air quality monitor AirPi. He says Kano takes a sense of engineering, creativity and problem solving to children, and presents it in a way that doesn't seem alien, so he expects even more youngsters to naturally join the community.

> "It's a really great time to be young because the tools are there"

"It's a really great time to be young because the tools are there," he says. "You can get a computer for $129 – a computer that you build yourself. The connectivity is there, you can log on and learn almost anything for free, whether through YouTube or Codecademy, and the enthusiasm is there. This is the *Minecraft* generation – they don't really care as much about the sweet graphics or the blazing fast performance because they have all of this on their mobile phones. When they go to create, what they're interested in doing is showing off. I have a brother who is 14 and he has built in *Minecraft* a scale replica of the Sienna Cathedral. Why did he do this? Not to make money but just because he saw other people doing something with this tool set and he wanted to basically say, 'Hey, look what you can do – I can do it too, maybe even better'."

Kano, he says, will harness this creativity by not bogging people down with unnecessary barriers. Alex feels that it is better to get people onto systems and using them first, and then encourage them to explore, rather than start from the bottom of the hill and work upwards – which, in a sense, is what the Raspberry Pi can feel like when a child gets hold of it for the first time. Much of it, he believes, is about confidence and allowing children to develop curiosity by simply steering them in the right direction.

"The original spirit of the Pi was for it to be a kind of code bomb, like a Trojan Horse that you would get for your kid and they would use to kind of become digitally literate, to start their journey on the road to becoming a technologist," he says. "We're creating a learning journey that I think makes that possible – although, of course, there's much more to be done."

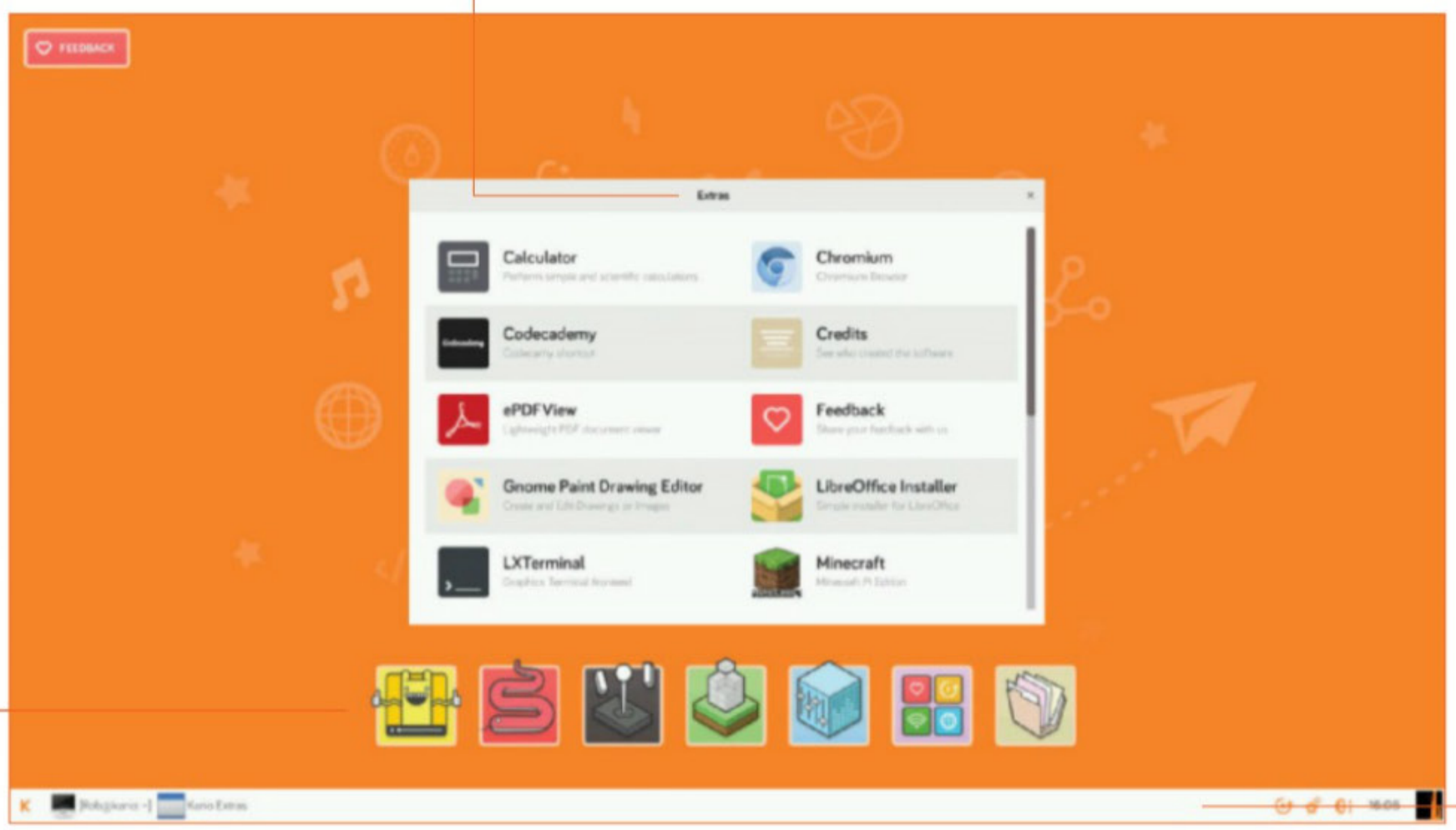
KANO

New features:
Completely custom interface • Introduction to the terminal • Coding challenges with Minecraft and Sonic Pi

Kids can use the teaching tools to get to grips with coding and computing with games and music

Kano can be used as a normal Raspberry Pi OS once you graduate from learning

Kano is a reskinned and customised version of Raspbian with great attention to aesthetics



DISTRO

Kano OS

Specs: Raspberry Pi Model B

Learning can be fun, especially when it uses Minecraft to teach coding and then allows you to do whatever you want afterwards

Pros
Excellent way to get children into real computing and coding while also being useful as a proper distro

Cons
Kano overclocks the Raspberry Pi to one step less than maximum and is still quite slow in some instances

The Kano computer system revolves around two core things: a Raspberry Pi and the Kano OS designed for it. More than just another Raspberry Pi kit, it proved itself with a successful Kickstarter, promising a system that would help children get into real computing and allow them to start down their own path of learning about programming and coding.

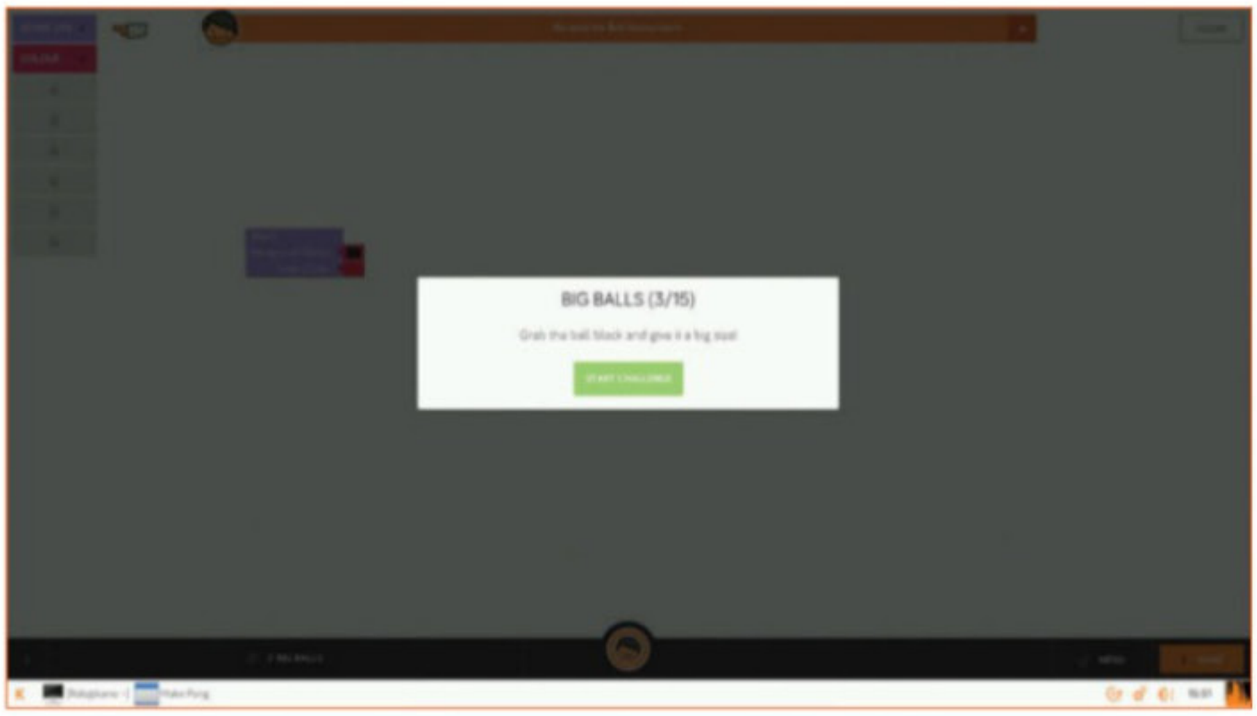
While the full kits are being prepared for shipping out to backers and other people that have pre-ordered, the beta for the full OS is available to anyone who wants it right now, completely free of charge. It doesn't require any of the specific hardware in the kit such as the Wi-Fi dongle or the wireless keyboard, so it will work on any normal Raspberry Pi.

With the OS, correct human inputs and a connection to the internet, you can get the full Kano experience. The first two books for the project are available for free and teach kids how to set up and use everything available in the Kano OS; they cover a lot of ground and the OS itself picks up the rest of the slack.

Make a computer

The installation and setup of a Raspberry Pi with Kano is an event in itself, covered entirely by the first book of the series. If you're on Windows or OS X, Kano provides tools to initially burn the image, however on Linux no such tools exist and you'll need to use the standard **dd** method. This is a little disappointing, as Kano itself is built upon Raspbian Linux and the team are very keen to promote FOSS in their operation.

Once written to disk, the true Kano magic starts for everyone. Turning on the Raspberry Pi for the first time has it 'speak' to you; a pre-determined script to set a username and a couple of key presses to make sure the keyboard is properly working,



The Kano guys have a bit of a sense of humour that probably won't go unnoticed by kids. And adults. Especially adults

Modify games with instant results; a great way to keep people engaged when they see changes take immediate effect

"Using Minecraft: Pi Edition is a truly excellent way to engage children"

and that you're paying attention as you follow the white rabbit. It's all very movie hacker – although without the Jonny Lee Miller crime angle – and it's engaging to younger kids who can be so influenced by media and pop culture. Following this and some very simple graphical setup tools, you're dropped into the main Kano desktop; it's all very quick to get here and afterwards you won't need to engage with the 'hacker script' again.

Learn to compute

From here you can start using the Kano as you would Raspbian, with full access to the default apps and some installers for LibreOffice and Wolfram if you need to do a little more than the basic apps provide. The main event though is the series of coding and computing challenges: making a video is the first one, a simple matter of making sure that your Pi is connected to the internet and typing YouTube; playing and modifying *Snake*, *Pong* and *Minecraft* to learn the basics of changing or modifying code; and making music with Sonic Pi to put it all together in actual written code.

Each of these comes with a set of specific challenges to complete that teach kids the intricacies of what they can modify and by how much. The *Snake* challenge is about giving different options to a terminal command, whereas *Pong* and *Minecraft* take a Scratch approach to building the code.

They're all very good at what they do and using *Minecraft: Pi Edition* for these challenges is a truly excellent way to engage children who cannot get enough of the game. The full version is also fully playable off the image, which is not the case with standard Raspbian.

Too good to be true?

There are a few caveats to Kano though – ones that will eventually bring people to standard Raspbian. The main issue we had with the OS is speed: browsing with the included Chromium browser brought the Pi to a halt and it took a long time for it to render Google or even the **Linux User** webpage. There is also the issue of the overclocking that Kano does to the Pi without even telling you about it; this will damage your Pi and at the very least shorten its life span in the long run – and Kano has it running at one setting below the maximum 1 GHz threshold.

Generally though this doesn't affect the actual teaching components of Kano: while there was a bit of slowdown in *Minecraft*, that was to be slightly expected. You can also turn down or turn off the overclock completely, while Midori is in the repos if you want to customise Kano a little more.

For now it's still an excellent teaching tool and a great way to get kids excited about programming and the Raspberry Pi, something no one else has been able to consistently capture.

Rob Zwetsloot

Summary

Kano is a brilliant tool for teaching and beginning the graduation to using the Pi as a proper computer. Although there are some speed issues to contend with, these may be ironed out for the final release and shouldn't deter you

Download now

kano.me/downloads

ROBOTS

PiBot: robotics for schools

Can building and programming robots help teach kids in schools?

The Raspberry Pi's primary purpose of teaching children is sometimes lost in the noise of what the excellent project hobbyists and makers are using the Pi for. The education angle seems to be making a comeback though, with projects like Kano appearing in this issue to remind us of that fact.

A new Kickstarter campaign has been launched in an effort to take a different approach to education with the Raspberry Pi: learning with robots. The PiBot is a project that hopes to teach kids physical and traditional computing by building and programming a Raspberry Pi-powered robot.

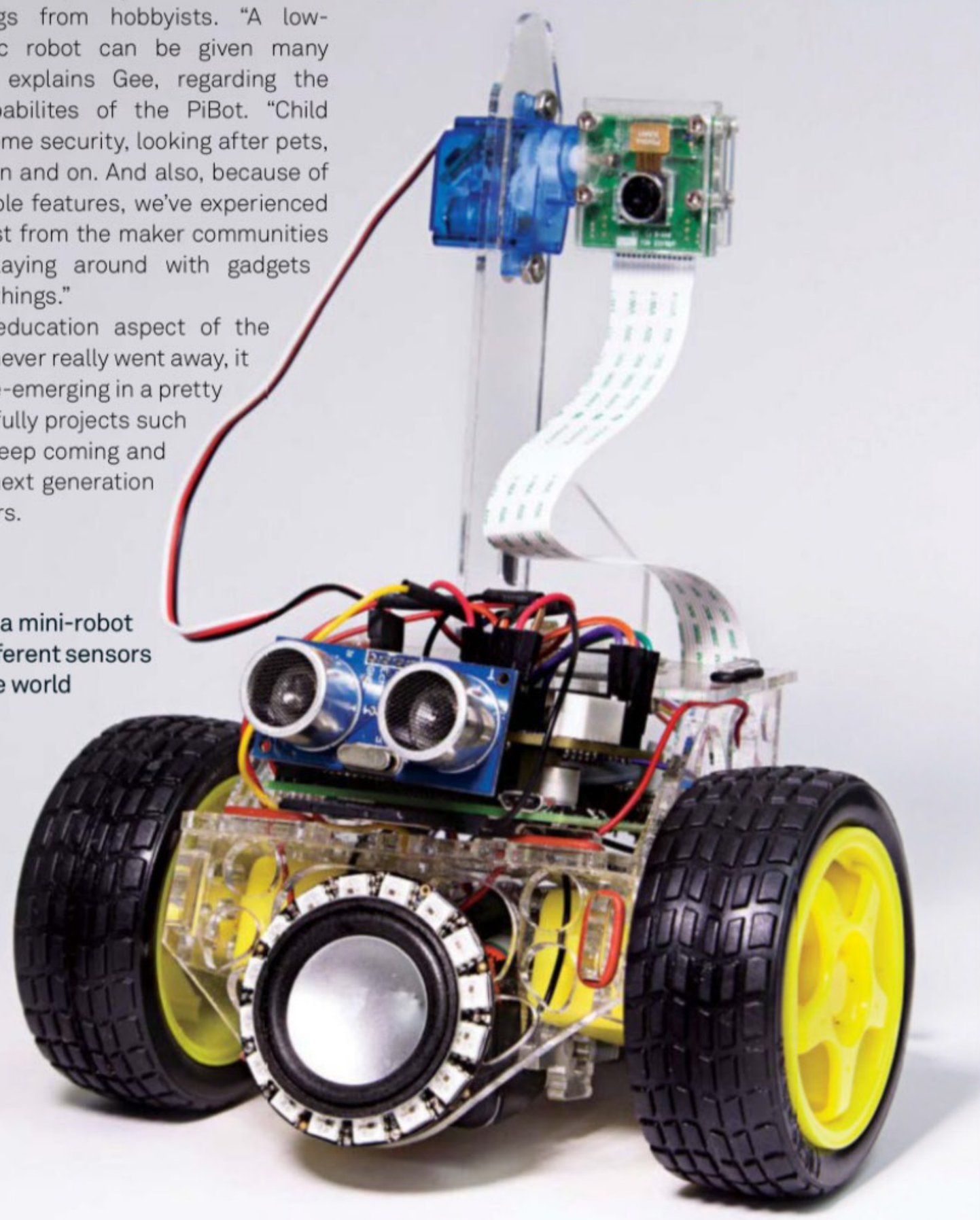
"The PiBot allows anyone to build, program and customise their own personal robot," Harry Gee, the PiBot project manager, tells us. "[It can help to teach many things] from how to program, problem solving, and why it's important to learn how to think. Now with technology changing, developing and progressing so rapidly, these sorts of skills will be very valuable in the near future. We have tested the PiBot in many workshops with kids and they have thoroughly enjoyed learning about robotics and the Raspberry Pi – and, of course, had plenty of fun."

While the Raspberry Pi itself is excellent as an education tool and has been used in many robot projects, why specifically has it been used by the PiBot team? Its popularity has helped according to Gee: "Since the Raspberry Pi launched just a few years ago, it has already established a great and diverse community. It already has hundreds of applications that can be used for robotics, eg voice recognition, face recognition, GPIO and so on. It is a device that enables people of all ages to explore computing and has the ability to interact with the outside world, giving programmers the ability to personalise the PiBot... Being affordable, customisable and available is so important and this helps push innovation."

While education is excellent, a lot of the existing Raspberry Pi robots have firm followings from hobbyists. "A low-cost domestic robot can be given many applications," explains Gee, regarding the extended capabilites of the PiBot. "Child monitoring, home security, looking after pets, the list goes on and on. And also, because of its customisable features, we've experienced a lot of interest from the maker communities who enjoy playing around with gadgets and inventing things."

While the education aspect of the Raspberry Pi never really went away, it seems to be re-emerging in a pretty big way. Hopefully projects such as the PiBot keep coming and inspiring the next generation of programmers.

> "The PiBot allows anyone to build their own robot"



■ The PiBot is a mini-robot with many different sensors to help see the world

Get involved

The Kickstarter for the PiBot will help this project come to fruition: you can help today by visiting the PiBot website at **pibot.org** to make a pledge and help the little robot teach kids in the future – you can even get one for yourself.

24/7 UK expert support

UK data centre

Dedicated to my business

99.99% uptime guarantee

Allowing businesses to do what they do best

Give your business the edge with Fasthosts range of Professional Series Dedicated Servers.

With 99.99% uptime guaranteed, SSD technology for faster performance, substantial storage drives and with **FREE** set up, there's a server to suit all budgets and business needs.

- **Full remote KVM control**
- **Up to 128GB RAM**
- **Unlimited bandwidth**
- **NEW cPanel and WHM**

Dedicated Servers from:

FREE setup

£39.00

Per month ex VAT
12 month minimum term contract

Call **0808 1686 777**
or visit **fasthosts.co.uk**

SERVERS · WEB HOSTING · DOMAIN NAMES · EXCHANGE EMAIL

fasthosts®

Inspiring Better Business

HARDWARE

Canonical unveils Orange Box cluster

Cloud-in-a-box kit built by Tranquil PC

At the OpenStack event in Atlanta, Canonical's Mark Shuttleworth unveiled the fruit of a secretive partnership with UK bespoke PC specialist Tranquil PC: the Ubuntu Orange Box, a clustered computing chassis that has been designed for cloud-related demonstration, education and experimentation.

The Ubuntu Orange Box is based around Intel's Next Unit of Computing (NUC) systems, packing ten D53427RKE motherboards into a single portable chassis. Each board includes an Intel Core i5-3427U dual-core processor with four logical HyperThreading cores, integrated Intel HD Graphics 4000 processors, 16GB of DDR3 memory, a 120GB SSD and gigabit Ethernet connectivity. These nodes are internally connected to a D-Link DGS-1100-16 managed switch, built into the same chassis. The first NUC node in the system is additionally given a 2TB mechanical hard drive, easy access to USB and HDMI ports and an Intel Centrino wireless adapter.

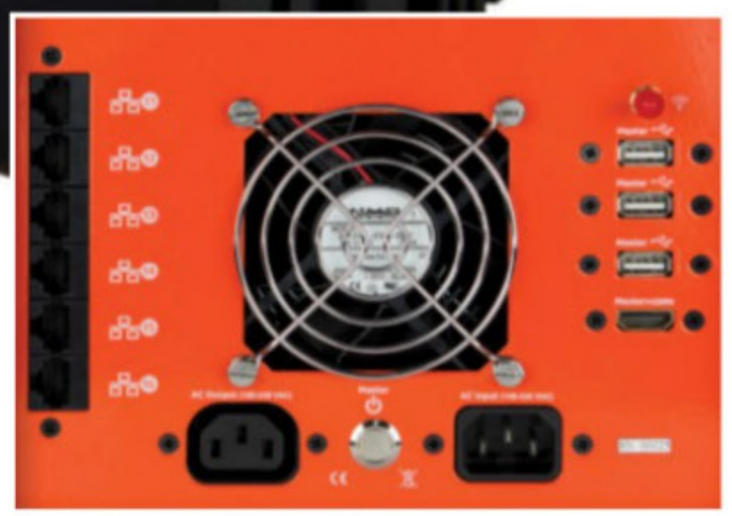
All told, the single-box cluster offers 40 logical processing cores, 160GB of RAM and 1.2TB of solid-state storage spread across ten nodes with gigabit connectivity. The spare six ports of the internal switch are broken out at the rear and allow for multiple Orange Box units to be easily clustered themselves for larger requirements.

Neither Canonical nor Tranquil PC is positioning the systems as server-room equipment. Instead, Shuttleworth presented the devices as being the perfect equipment for research and development, experimentation and education. The Orange Box units will also form the heart of the company's new Jumpstart Training, a two-day course which uses the cluster to help teach Ubuntu, Metal-as-a-Service (MaaS), Juju, Landscape and OpenStack administration.

The systems are also being sold by Tranquil PC directly, however: pricing starts at £7,575 for the base model, with additional storage being available for each four of the ten nodes. For those planning to take their Orange Box units on tour, a £345 Peli Case option is also available to ensure safety during transit.

"You can do anything with these Orange Boxes," claimed Canonical founder Shuttleworth during his keynote speech at OpenStack. "These are a great way to learn how to use distributed systems."

Impressively, given its specifications, the Orange Box is designed to be cooled by a single fan located at the rear. The processor of each node is from Intel's reduced-power family, meaning it can be cooled using a passive heatsink. This makes for a surprisingly quiet cluster – which could well help increase its appeal for use in classrooms.



■ The Ubuntu Orange Box packs in ten NUC-based cluster nodes, has 40 logical processing cores and 160GB of RAM

Speaking to press after the presentation, Canonical staff indicated that demand for the devices was high, despite the company's plan having originally been to use the Orange Box exclusively for Jumpstart Training sessions. The company is reportedly investigating how to cope with international demand from those who are looking to put the many-core clusters into immediate production use.

HARDWARE

Mozilla lights Flame reference handset

■ The Flame represents the first reference design for a Firefox OS handset

The Mozilla Foundation has launched its first reference handset for the Firefox OS platform, aimed directly at developers.

A more mid-range device than the distinctly entry-level ZTE Open and its successor the ZTE Open C, the Flame boasts a 1.2GHz dual-core Qualcomm MSM8210 Snapdragon system-on-chip processor, 1GB of RAM, 8GB of storage and support for micro-SD card expansion, all behind a 4.5-inch 480 x 854 FWVGA capacitive display.

As well as offering higher performance than existing handsets, the Flame includes a neat feature for developers: the ability to artificially restrict system RAM down to as low as 256MB, as a way of testing out the performance of their applications on the lower-end handsets.

The device is being sold for $170 with free international shipping, in a black finish only.

SECURITY

Core Infrastructure beneficiaries revealed

NTP, OpenSSH and OpenSSL the chosen few

The Linux Foundation has announced the projects that will benefit from funding and support under its security-enhancing Core Infrastructure Initiative: the Network Time Protocol, OpenSSH and OpenSSL.

Each project, the Foundation has explained, will receive funding in order to conduct code audits and security analyses in the wake of the Heartbleed vulnerability that left numerous OpenSSH-based servers vulnerable to private key disclosure attacks. OpenSSL itself will receive enough funding for two full-time core developers, while further funds will be provided to the Open Crypto Audit Project – founded to analyse the security of TrueCrypt prior to its apparent demise – for a full security audit.

The Foundation also announced new founding members for the Core Infrastructure Initiative itself: Adobe, Bloomberg, HP, Huawei and Salesforce are all now officially part of the project, providing funding for the Steering Committee's disbursement.

"CII implements the same collaborative approach that is used to build software to help fund the most critical projects. The aim of CII is to move from the reactive, crisis-driven responses to a measured, proactive way to identify and fund those projects that are in need," reiterated Foundation director Jim Zemlin. "I am thrilled that we now have a forum to connect those in need with those with funds."

■ Linux Foundation executive director Jim Zemlin has said he is "thrilled" with the Core Infrastructure Initiative

Linux calendar

1st – 2nd July

SophiaConf

» Polytech Nice Sophia, Cedex
» France
» sophiaconf.fr

Organised by la Commission Open Source de Telecom Valley, SophiaConf's programme will have talks about the semantic web, cloud security, the Internet of Things and more.

20th – 24th July

Automotive Linux Summit

» Chinzan-so Conference Centre, Tokyo
» Japan
» events.linuxfoundation.org

Designed to further the use of Linux and associated technologies in vehicle computer systems, the official Automotive Linux Summit includes speakers from around the globe.

20th – 24th July

OSCON

» Oregon Convention Centre, Oregon
» USA
» oscon.com

O'Reilly's annual Open Source Convention, OSCON this year includes noted speakers Tim Bray, Will Marshall, Jono Bacon, David Baumgold, Erik Rose and dozens of others as well.

20th – 26th July

Linux Bier Wanderung

» Henderson Hall, Talybont-on-Usk
» Wales
» lbw2014.bp.be

An annual event, LBW will be taking place in Wales this year with a range of talks, hands-on projects and outdoor activities by and for open source enthusiasts from a number of nations.

SECURITY

TrueCrypt declared insecure, project halted

Security analysts continue their investigation

Popular open-source cryptography project TrueCrypt was brought to a shuddering halt when its creators deleted it from its SourceForge repository, replacing it with a message warning against its use.

The official repository has had all previous source code and binary builds removed and replaced with a decryption-only tool dubbed TrueCrypt 7.2. This software includes the cryptic warning: "Using TrueCrypt is not secure as it may contain unfixed security issues. The development of TrueCrypt was ended in 5/2014 after Microsoft terminated support of Windows XP. Windows 8/7/Vista and later offer integrated support for encrypted disks and virtual disk images. Such integrated support is also available on other platforms. You should migrate any data encrypted by TrueCrypt to encrypted disks or virtual disk images supported on your platform."

The lengthy and odd wording of the message, the sudden nature of the move and the rather strange recommendation to move to a proprietary closed-source alternative has left many people wondering whether the maintainers of the project have received a National Security Letter (NSL) from US law enforcement demanding back-door access be placed into the code, or whether an existing back-door vulnerability has been discovered.

The latter could come to light as part of the Open Crypto Audit Project, which had recently completed the first phase of a crowd-funded security analysis of the software with no critical issues found. The researchers behind the project have pledged to continue with their investigation and, if no back-door access is found, potentially fork the project as of version 7.1 using previously released source code.

OPEN SOURCE

Nouveau now supported by Gallium3D

Nvidia has made another significant contribution to the Nouveau open-source graphics driver project, this time adding Gallium3D support for its Tegra K1 system-on-chip processor.

One of only a handful of contributions Nvidia has made to the project since its inception, the move is important for the company owing to the launch of the Tegra K1-based Jetson single-board computer (SBC), with which the company is hoping to capture the hearts and minds of the hacker and maker communities.

The move provides support for accessing the powerful Kepler graphics processing units (GPUs) built into the Tegra K1, which offer 192 stream processing cores for highly parallel tasks.

GAMING

Valve Steam Machine delayed to 2015

Dedicated Steam Controller redesign to blame

Valve's Steam Machine, its first foray into the realm of hardware and the impetus behind its decision to port its Steam digital distribution platform and first-party games to Linux, has been officially delayed to next year.

Originally scheduled to launch later this year, Valve's official Steam Machine was to run a customised Debian Linux distribution with a focus on gaming called SteamOS. The machine, a console-like creation using off-the-shelf PC hardware, was to come bundled with a specialised controller designed to compete with a keyboard and mouse for speed and accuracy – and it's here that Valve appears to have come unstuck.

While the Steam Controller's clever touch-sensitive thumb surfaces have remained intact, the design of the controller overall has changed dramatically. Plans to place a touch-sensitive display at the centre have been scrapped in favour of a dual-diamond button layout and the company has warned that feedback from users of prototype devices means there are more changes to come.

Warning fans of the delay, the company claimed it was now looking at a release some time in 2015 for the Steam Controller – and, by extension, the official Steam Machine. The move is not thought to affect third-party Steam Machines (which license the name and concept from Valve); these are still on course to launch later this year without the specialised Steam Controller.

Valve has yet to confirm pricing or specifications for the official Steam Machine, prototypes of which were supplied with high-end graphics cards and processors but the final design of which is likely to rely on more affordable components.

■ Valve's Steam Controller is undergoing another redesign, delaying the Steam Machine to 2015

EDUCATION

Linux Foundation, edX team up for beginner's course

Free education site offers introductory Linux training

The MIT- and Harvard-founded edX not-for-profit education site edX has launched a new free course, supported by the Linux Foundation, offering an introduction to the open-source operating system.

Aimed at those with a general familiarity of computers but no direct experience of using Linux, the course is claimed to take around 40 to 60 hours to complete and includes an introduction by Linux founder Linus Torvalds. Topics covered include the tools and techniques required to use Linux as an end-user, a system administrator and finally a programmer.

According to Jerry Cooperstein, Linux Foundation training programme director and the creator and maintainer of the Introduction to Linux edX classes, those running through the course will finish up with a working and distribution-agnostic knowledge of how to use Linux from the command line and via a graphical user interface (GUI).

Although the course is free to attend as an 'auditor' – which provides complete access to the lessons, tests and the web-based discussion forum – edX also offers what it describes as a Verified Certificate of Achievement, providing attendance through to completion for a fee.

The course begins on 1 August, with details available at **edx.org** under course code LFS101x.

The edX Introduction to Linux course comes with Linus' official seal of approval

OPEN SOURCE

New York Council receives Open Source bill

Ben Kallos looks to mandate a FOSS preference in government

New York City Council member Ben Kallos has introduced a bill dubbed the Free and Open Source Software Act that asks the Council to mandate a preference for FOSS over proprietary software when putting work to tender.

An open source developer himself, Kallos has told local media that the government was behind the times in appreciating the cost savings available simply by switching from proprietary software to open source alternatives – both in terms of initial capital expenditure and total cost of ownership.

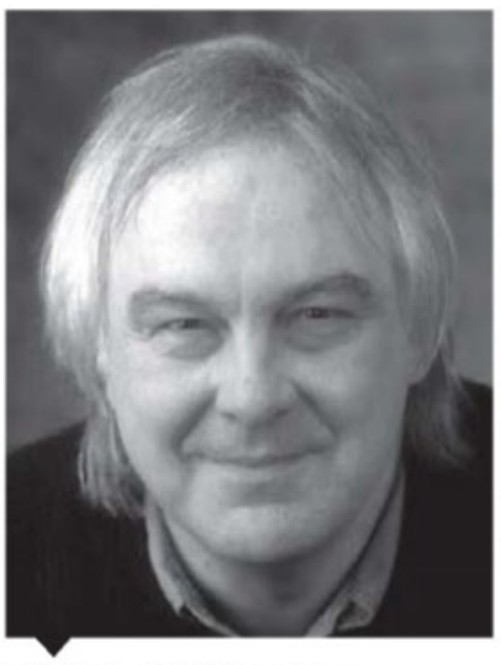
If passed, New York would join San Francisco and Oklahoma in the list of US councils that have mandated the use of open source software by preference with the purchase of proprietary software only available where no reasonable FOSS alternative exists.

The bill will be read before the council, and a vote taken as to whether to pass the matter into law. At the time of writing, it was not clear how the vote would go – but those arguing against it may have a tough time, given Kallos' experience and passion.

THE FREE SOFTWARE COLUMN

Just for fun

In these corporate times, it can be easy to forget that Linux was originally written 'just for fun'. The word 'hack' originated as an expression of the idea of adding value with a quick fix – and hacking was fun



Richard Hillesley writes about art, music, digital rights, Linux and free software for a variety of publications

Originally, a 'hacker' was 'someone who made furniture with an axe'. Transpose the image to coding and a hacker becomes someone who knocks code into shape with speed, making something from it that is both elegant and useful. In the language of the early hackers of the AI Lab, as recorded in the 'jargon file' (**http://en.wikipedia.org/wiki/Jargon_File**), a hacker was 'a person who enjoys exploring the details of programmable systems and how to stretch their capabilities, as opposed to most users, who prefer to learn only the minimum necessary' (**ww.catb.org/jargon/html/H/hacker.html**)

A hacker was also described as 'one who programs enthusiastically (even obsessively) or who enjoys programming rather than just theorising about programming'. The hackers of the Tech Model Railroad Club (TMRC) and later the Artificial Intelligence Lab at MIT saw programming not as a job but a creative adventure – a chance to stretch their horizons.

Programming wasn't just a means to an end, but a source of creativity and fun, a subversive path to an elegant solution. One of the very early definitions of the word 'hack' was: 'to explore the basements, roof ledges, and steam tunnels of a large, institutional building, to the dismay of Physical Plant workers and (since this is usually performed at educational institutions) the Campus Police. This activity has been found to be eerily similar to playing adventure games such as Dungeons & Dragons and Zork' (**www.lysator.liu.se/hackdict/split2/hack.html**).

The hackers of the TMRC were also known as The Midnight Requisition Committee, in recognition of their sometimes illicit midnight forays through the corridors of Building 20 to collect electrical bits and pieces for the layout tables, or Building 26 to steal time on the TX-0 computer late at night. As recorded in Steven Levy's *Hackers: Heroes Of The Computer Revolution*: "When no person in his right mind would have signed up for an hour-long session on the piece of paper posted every Friday beside the air conditioner in the RLE lab... the TMRC hackers, who soon were referring to themselves as TX-0 hackers, changed their lifestyle to accommodate the computer."

These hackers developed the first Lisp machine, the first computer games, the first music software and the first display hacks. The limits and potential of their programming activities were described by The Hacker Ethic. (**http://en.wikipedia.org/wiki/Hacker_ethic**). Among hackers, programming was elevated to something like an art form. As Michael Cardell, a later advocate, tells it: "A true hacker is generally able to appreciate what is called 'hack value', a mysterious entity found in great works of art, great theories, workmanship, music and, of course, computer programs" (**http://hack.org/mc/writings/scriptures.html**).

This spirit of playfulness and creativity was inherited by GNU and the free software movement as free software

evolved through the Eighties. Free software was idealistic and serious, but always had elements of playfulness too, reflected in a compulsive use of wordplay. Stallman's humorous adoption of the persona of St iGNUcius (**https://stallman.org/saint.html**) and the Church of Emacs (**http://www.emacswiki.org/emacs/ChurchOfEmacs**) is itself reminiscent of the tradition of the hacker koans (humorous short stories about computer science) of an earlier era.

When Linux arrived in the early Nineties, Linus Torvalds made the fortunate decision to release the Linux kernel under the GPL and to announce his project on Usenet, which made it easy for others to contribute and participate. The arrival of Linux happily coincided with the arrival of the World Wide Web and the explosion of the internet. The community that grew around GNU and Linux during its first decade was dominated by hobbyists and students, and was dedicated to experiment and fun, impossible dreams and improbable success. Very few kernel hackers were able to make a living directly from their work on the kernel until the turn of the decade, when the UNIX companies began to realise the latent value of Linux and free software and the advantage of employing those that had made it possible.

The kernel, and other components of the free software stack, are now seen as the key elements of an increasingly diverse array of corporate product lines, which may be a good thing. Free and open source software has opened opportunities to everyone, but the spirit and playfulness lives on in the community distributions and non-commercial projects, which are dedicated to expression and exploration of ideas. Hobbyists still have a role in keeping software alive and free.

THE OPEN SOURCE COLUMN

Bad coffee at 30,000 feet

Simon ponders the technology of the consumer aviation industry as he taxis down the runway at Birmingham Airport...

Simon Brew is a technology writer and editor, working across the Linux, Windows and Mac OS X platforms

Had the pleasure of a United Airlines flight of late? For reasons that fall firmly below the line marked 'interesting', I've ended up on a couple of them of late, shoved to the back of the cattle section, grasping at anything that looked like a cup of coffee as it shuffled up the aisle on a trolley.

In days gone by, when you were on a flight, the safety demonstration was done by human beings. But human beings – as the self-service checkouts at any supermarket will demonstrate – are a disposable currency, it seems. So now you get a fluffy video, which in United's case is pre-empted by some people in suits generally telling people how awesome they are. I'm paraphrasing, because the coffee hadn't turned up by this point.

But one of these posh people got my interest. Because he started talking, as my flights were taxiing to the runway, of how his job was to ensure that United customers have the very best mobile applications. Never mind that something useful like Wi-Fi in the sky wasn't being offered, instead he demonstrated tools that helped me check in at the airport and access my boarding pass and a seat map on my ageing phone.

And, to be fair, some of these tools have proven very useful. My initial temptation to write something grumpy about all of this, and about how software is supposed to complement people rather than replace them, was resisted when it became clear that United – as many airlines do – had come up with some interesting stuff.

Yet I can't help drawing parallels to USB chargers for mobile phones. One of the very first columns I wrote for this very magazine was bemoaning the fact that major companies had subverted well-established USB technology to come up with a variety of ill-shaped ports to facilitate the selling of an expensive charging system. That's not quite the case here, but the whole idea of unifying a check-in, booking, boarding pass and seat map system surely has benefits.

Sadly, we're in an era where it's not just tools like these that are individual to each airline, but also where different websites from the same company tell you different things. The abuse of cookies is another familiar drum to beat. In the aviation industry, it's long been proven that if you delete your cookies and flush your cache, you can sometimes get a better deal by going to an international version of the same company's website. In some cases, you can save hundreds of pounds on the same trip just by booking it in another country, with the same firm.

See, that's the kind of thing that baffles me. And as much as I salute the way technology has tried to make the inconvenience of battling airports as convenient as possible, there are so many inconsistencies, as a consequence of so many companies going against each other. Is it idealistic to suggest that, at the very least, individual airports could impose some consistency amongst the airlines they play host to? And to call for using said technology for more transparency on pricing?

It's the ongoing irony of technology: the more things advance – the more tools there are that allow us to do more things – the easier it seems to be to hide stuff from us.

The coffee was rubbish too, if you're interested.

JON MASTERS

The kernel column

Jon Masters covers the latest goings on in the Linux kernel community, as the development on the 3.15 kernel comes to a close and 3.16 work begins

Jon Masters is a Linux kernel hacker who has been working on Linux for some 19 years, since he first attended university at the age of 13. Jon lives in Cambridge, Massachusetts, and works for a large enterprise Linux vendor. He publishes a daily Linux kernel mailing list summary at **kernelpodcast.org**

Linus Torvalds announced the final Release Candidate (RC) for what will become Linux 3.15, noting that he felt pretty comfortable with the state of things at this point. The 3.15-rc8 kernel contains just a smattering of core kernel fixes (some in the scheduler, some in the filesystem code), and a few more architecture-specific patches, but relatively little overall in the way of churn. In other words, 3.15 is largely baked and ready to go, with the weekly RCs serving their purpose of gradually tapering off toward the final RC7 or RC8 release. Oftentimes, final Linux kernels are released following the RC7 timeframe, with no need for an RC8 to be issued, but on this particular occasion there was enough in the way of small last-minute fixes for Linus to feel justified in holding off another week with an RC8 instead.

As Linus notes, "Normally, an RC8 isn't really a big deal - 3.15 is one of the biggest (if not *the* biggest) releases in a long time, and we do RC8s with some regularity". It's true that, as he goes on to say in his announcement, roughly half of the kernel releases have an RC8, where the other releases go out after RC7. The concern Linus has with having an RC8 for Linux 3.15 is more elementary: he's going on vacation, and he will be travelling enough over the coming weeks that he didn't want another merge window during which there was any potential for disruption due to his travel plans. As a consequence, Linus is trying something new for the 3.16 kernel cycle. Rather than wait a week for the final 3.15 to go out before opening the 3.16 merge window for new 3.16 features to land in his development tree, Linus decided to open the 'merge window' concurrent with the 3.15-rc8 release and see if he could streamline the process a little bit more.

In order to achieve a concurrent release of 3.15-rc8 alongside the opening of the merge window for 3.16, Linus is using a '-next' git branch (similar in name to the branches used by his maintainers to stage bits for the next kernel cycle while the previous cycle is ongoing) in his source code repository, in which he is staging the bits that will be pulled in for Linux 3.16-rc1 while leaving his standard 'master' branch to carry the final 3.15 patches ahead of the 3.15 release. If things work out, he will release 3.15 and move the 3.16 bits over into the usual place (the master branch), saving a week of productive time at a point in the kernel development cycle where the risk of disruption is pretty low. It is an interesting experiment, and if it works out, perhaps will be repeated in future cycles.

16K kernel stacks

One of the 'fixes' that did make it into Linux 3.15 after all was the switch to a 16K kernel stack on 64-bit x86 systems. This 'one liner' from Minchan Kim was called out in Linus' 3.15-rc8 announcement as doing "something I've been trying to delay for a long time". Linux, like many other operating systems, uses a separate (fixed size) kernel stack for every task (known as a 'process' to users) in the system. This stack is statically allocated for each task and is used to store the kernel context whenever a task makes a system call (or an interrupt occurs) that causes the kernel to do something on behalf of an application. The kernel context essentially includes local data being used by functions within the kernel that are servicing the system call request at a given moment. Depending upon how deep the call chain (how many functions call one another) goes, this stack can come close to being exhausted. If it is exhausted, the kernel will experience a fault that will minimally kill whatever task is running, but will likely also result in a system crash immediately, or within a short space of time.

Over the years, the kernel has become increasingly complex, to the point that the original 4K, and later 8K, stack was no longer sufficient to handle the possible level of nesting of function calls – especially when performing some complex file system operations on network storage or other layered protocols (the XFS filesystem particularly suffers when trying to live within a smaller stack). Theoretically, the level of nesting could result in very large amounts of memory use, but in practice, this doesn't occur and Linux still largely gets away with the 8K choice on x86_64. But enough developers now feel strongly that the stack is insufficiently sized that it will be increased in 3.15 to avoid potential problems.

You might wonder how the kernel could run out of stack space. After all, applications don't have the same problems, right? It turns

out applications don't have to deal with a fixed size stack because their stack is automatically extended by the kernel on demand. Whenever the userspace stack for a given task ('process') reaches its limit (because many nested functions were called, or because a large amount of data was allocated on the stack through local functions – programmers today don't even think about this), the subsequent page fault that results from the application attempting to access beyond the current stack limit is trapped by the kernel, which allocates more stack to the process.

This cannot be done within the kernel itself however. Kernel stacks consume a fixed amount of (unpageable – it cannot be 'swapped out' to disk) memory. These need to be linear in physical memory since they may be used in cases where that matters, and consequently cannot be too large to avoid the additional complication of finding available chunks of contiguous linear physical memory. For these reasons, the stack size is kept as small as it can possibly be, but it was long since time for x86 to catch up with the other architecture stack hogs that had long since also moved to 16K.

Ongoing development

Borislav Petkov posted a patch entitled 'CPU hotplug: Slow down hotplug operations' that introduces a delay in the onlining and offlining of CPUs within a hotplug system. He claims this is because many of those running tests are creating implausible scenarios with hundreds or thousands of continuous offline/online operations, and this is exposing fundamental problems with the hotplug code. His sentiment was respected, but the consensus was that hiding such problems won't make the situation better. Thomas Gleixner noted that many had proposed they would work on cleaning up hotplug and that he hoped to focus on this again soon.

Alex Williamson posted a 'new device binding path using pci_dev.driver_override' patch that extends the existing support in the kernel for dynamically 'binding' and 'unbinding' drivers to devices by adding support for telling the kernel which driver should bind to a given device in the case that there is more than one driver loaded that could provide support for a piece of hardware. The new sysfs entry found in /sys/bus/pci/devices/.../driver_override allows for fine-grained control in the case that a single device must be bound to a meta driver within a virtualised environment.

Christoph Hellwig announced a new scsi patch queue tree to help out James Bottomley by staging many of the SCSI patches that have been pending for upstream. The new tree is split into SCSI core patches and those of drivers. It is available from the **git.infradead.org** site under users/hch/scsi-queue.git and comes with a series of rules of engagement that were posted to the Linux kernel mailing list.

Finally, Ted Ts'o posted a reminder that the nomination process for the 2014 Kernel Summit is open, with those wishing to attend able to propose topics. Kernel Summit typically draws about 100 of the core developer kernel community, but also aims to be inclusive of the broader community. Those with a topic worthy of consideration can consult online archives of the relevent mailing list: **kernel-summit@lists.linuxfoundation.org**.

"Over the years, the kernel has become increasingly complex"


Computer
Home
All Applications
Accessories
Graphics
Internet
Office
Sound & Video
Preferences
Administration
Places
Recent Files
Firefox Web Browser
HexChat
Pidgin Internet Messenger
Thunderbird Mail
Transmission
Firefox Web Browser
Browse the World Wide Web
Internet
E-mail
Menu
All Programs
Log Off
Turn Off Computer
start
Control Panel
My Documents
/ISUS

Switch to Linux

Make the jump from the ageing Windows XP to a Linux distro with our complete guide and you'll never look back

The inevitable has happened; the once-revered Windows XP has been taken out back by Bill Gates and tearfully shown the business end of a shotgun after many years of loyal service at the Redmond ranch. The end of security support is the final nail in the coffin from Microsoft as it tries to shuffle its expansive client and customer base over to Windows 8 if it can – or Windows 7 if it must– before the gun has even finished smoking from the deed.

Microsoft would make you think it's the only alternative, however that ignores the shining beacon of Linux just beyond the horizon. Once thought to be the malformed operating system of only the most hardcore tech nerds, speaking in riddles and snake languages such as 'Python', the Linux landscape has changed to be more welcoming to everyone.

Spurred on by the success of Ubuntu, the various versions of Linux – known as distributions or distros – have become more stable, more usable and generally more useful for your day-to-day computing. Over the next few pages we'll help you safely make the switch and begin your enlightening journey into the world of Linux. Let's get going!

Keeping XP

We'll let you know various methods for keeping or being able to return to XP if you need to at various points in this tutorial. Linux is all about choice, and you can choose to run it with XP if you wish.

Resources

Having problems with your first forays into Linux that Google can't solve? These people will be able to help...

forums.linuxmint.com
The Linux Mint forum is an active community of people that has answered many questions on all sorts of Mint topics. If you can't find your specific problem, make a post and they will do their best to give you advice.

LinuxQuestions.org
This is the nexus of all Linux knowledge, with a setup dedicated to helping people and a good search system to make sure you're not repeating a question. They'll answer all types of questions for all the software you'll be using and any distros you might find yourself using in the future.

If all else fails, feel free to get in touch with us on Facebook (**Linux User & Developer**) or Twitter (**@linuxusermag**)

Backing up

Get familiar with the process of backing up your important files and documents before making the switch

If you're a well-organised person you will very likely already have your PC or laptop backed up, which means a gold star for you. Well done! A lot of us unfortunately forget to get a backup solution sorted out, though. If you're at this stage, it doesn't really matter so much about creating a backup plan for your Windows machine but you will need to create copies of your files, documents and anything else you want to take over with you to Linux.

What this involves is very simple: all you have to do is create a copy of the data in your My Documents folder. This folder holds your actual documents, your videos, your pictures, your music and just about any other personal file of yours. Now may also be a good time to have a quick scan through the folder to see if there's anything you don't specifically need – have a spring clean of your files, if you will. Once that's done, you can copy them to a place off of your computer; we suggest doing this onto a spare USB stick or external/portable hard drive so you can easily get them when your new Linux is all set up.

Very few of your programs will need backing up; they'll either all be replaced or work so differently on Linux that there would be no real point. If there's a way to export a profile, like on Firefox or another program you know you'll be able to use on Linux, get that now and add it to your collection of backed-up files.

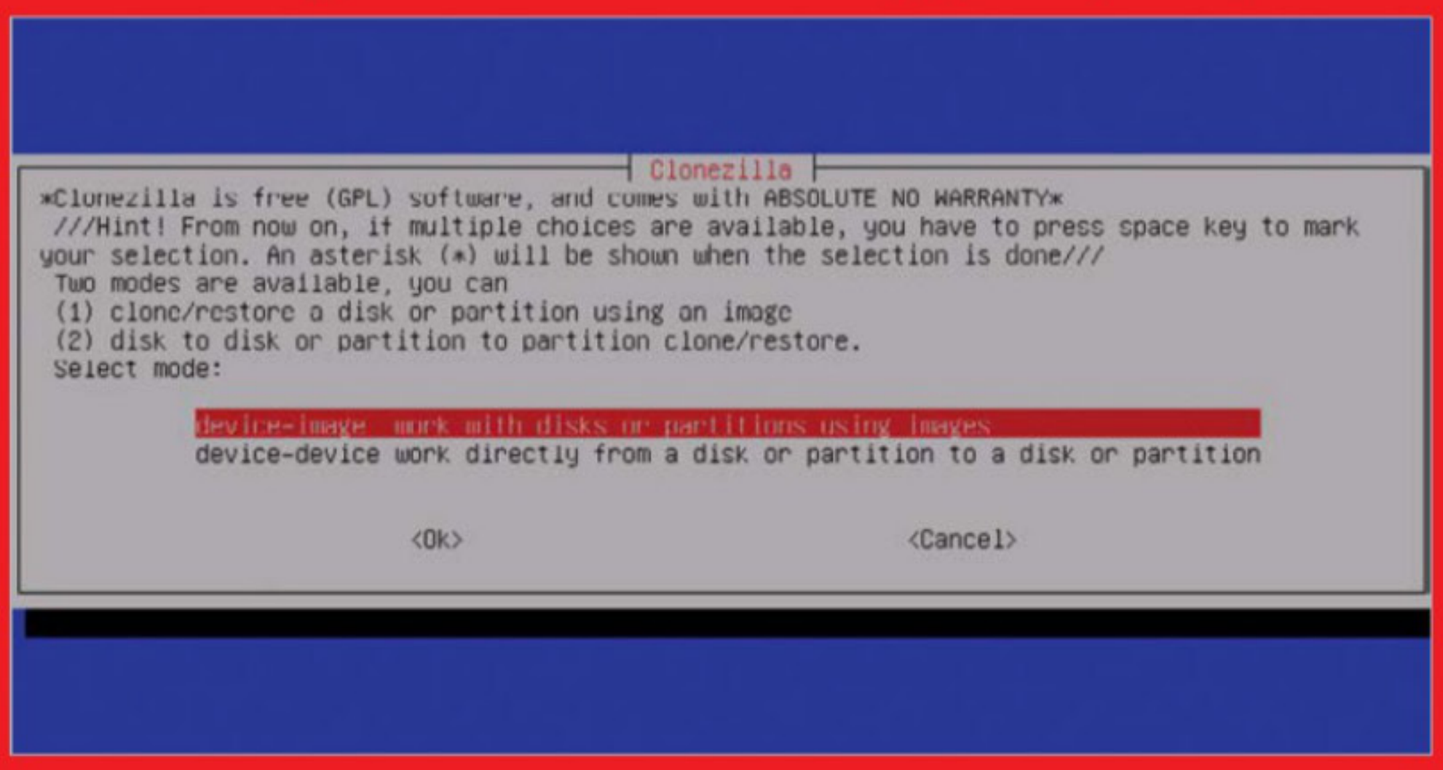
Full backup

We've talked about saving the files and documents that you need the most and putting them in a safe place, but if you want to do a proper backup just in case, we recommend using the excellent Bacula (**blog.bacula.org**). It's incredibly powerful, open source and also available on Linux, so you'll be able to use it on there for backups once you're all settled too.

Clone your hard drive

As well as backing up all your files, you can make a perfect copy of your hard drive in case you ever want to reinstall it to precisely how it was. It's also a sure-fire way to restore the computer to exactly the way it was before you installed Linux. The Linux distro Clonezilla (**www.clonezilla.org**) can boot directly from a CD without needing to be installed and can create an 'image' of your hard drive, which is information on all of the data – where it is, the size of it, and so on. This image can be used with Clonezilla to restore the hard drive as well.

Make sure you have sufficient storage space though, as the image will be as large as your hard drive. A big external hard drive will do the trick as long as you can spare the space.



Virtualise your hard drive

A more novel – yet still incredibly useful – way to keep a copy of your old PC is by creating a special virtual hard drive with all the data currently on it. Microsoft actually supplies a pretty helpful tool to help you achieve this using your XP computer called Disk2vhd (available at **bit.ly/18b901i**).

These snapshots can be created onto the computer you're already using, however it's a little quicker to simply save it straight to an external hard drive. Once you've got Linux installed and ready to go, you can always create a 'virtual machine' with Virtual Box using this vhd file, resulting in a fully working version of your original computer setup inside your new computer setup. This can be especially handy if there are some specific programs that you want to be able to use on Windows, or if you forgot to back up some specific files earlier on.

We recommend...

Discover some of the best distros on offer that will help ease you into the world of Linux

Also consider...
Ubuntu • Lubuntu • ElementaryOS
CentOS • wattOS • Debian • Arch

Linux Mint

Pros
Designed around the user, Linux Mint is clean, fast and straightforward. Being built on top of Ubuntu means it has access to a lot of software

Cons
For old PCs this may not the best option, but there are alternative versions of Mint with desktop environments that will work better

More info:
linuxmint.com

We'll be using Linux Mint as our example distro during this guide. Based on the popular Ubuntu distro, it strips away the more commercial features and adds its own 'desktop environment': the term used for the graphical interface you use to interact with the computer. The excellent Cinnamon desktop is reminiscent of Windows XP, yet has the search functions of newer distros and a more customisable program bar at the bottom. It's incredibly easy to use.

Fedora

Fedora Linux is famous for being the test bed for Red Hat Enterprise Linux, one of the most profitable and popular versions of Linux around. Fedora is strict with what it includes for you to install and download, preferring open source software. This means you may have problems playing MP3s and a lot of video files unless you modify it using RPM Fusion. It's still a great look into the most up-to-date Linux software around though, and many people use it as their main distro.

Pros
On the cutting edge of software and well supported by its creators, along with access to many apps and programs

Cons
Uses the GNOME desktop, which is quite different to Windows XP and not well regarded in the wider Linux community

More info:
fedoraproject.org

Mageia

Pros
Very easy to use and is designed around the user more than anything. The control centre is also extremely well designed

Cons
Some quirks when installing if you do it slightly the wrong way and it may not be as well supported or popular as some other distros

More info:
mageia.org

A relative newcomer to the Linux scene, Mageia is a wonderful operating system to use that is possibly one of the most accessible versions of Linux we've used. It has a few quirks during initial setup that may be a little beyond the most novice users but it can be well worth it in the long run. It also has an excellent control centre like openSUSE and can use similar installation files like the ones found in Fedora. If you're looking for something after Linux Mint, this is it.

openSUSE

A community-run project that is to SUSE – an enterprise Linux operating system – as Fedora is to Red Hat Enterprise Linux. It's less cutting-edge and is more optimised for being used at home or in the office, though. OpenSUSE has an excellent settings program that allows you to easily control all of the major functions of the operating system in great detail. The project's goals are to make it the most accessible Linux in the world – and it's well on its way.

Pros
Community supported and uses slightly more stable and known software as Fedora, without any of the open source restrictions

Cons
Not the most popular distro in the world and uses quite different installer files compared to the other distros here

More info:
opensuse.org

Install Linux

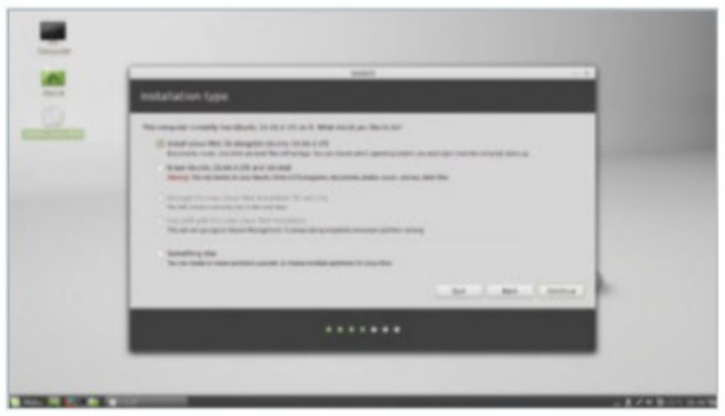
Learn the right way to install Linux Mint and how to get the most out of it

Once you've got your Linux Mint image downloaded (or other distro if you fancy using a different one), you'll need to burn it to a spare DVD or temporarily create a bootable USB stick with it. We recommend doing the latter by using the UNetbootin software (**unetbootin.sourceforge.net**) and a spare USB stick that's at least 2GB in size. Be sure to back up any files on the USB stick before using the software though, as it will delete them otherwise.

Once that's all been dealt with, simply reboot your computer with the disc in the tray or USB stick still attached and look out for the 'boot menu' key when your computer first turns back on – this will probably be something like F12 or another function key.

If you want to be able to keep Windows and install Linux Mint alongside it, refer to the information on shrinking in the boxout at the bottom of this page.

■ The first screen you'll see when live-booting Linux Mint – this means you're doing it right!



01 Hard drive

Follow the installation instructions on the screen. Here you can choose whether or not to keep XP; look to the boxout below for information on how to install alongside, however if you want to remove XP you can just select Erase Windows XP before hitting Continue.

02 Username

You'll need to set a username and password for your computer. Don't worry about the name of the install – keep it simple. Before clicking Next you can choose between automatic login or putting in a password manually – think about where your PC will be and choose accordingly.

03 Let it install

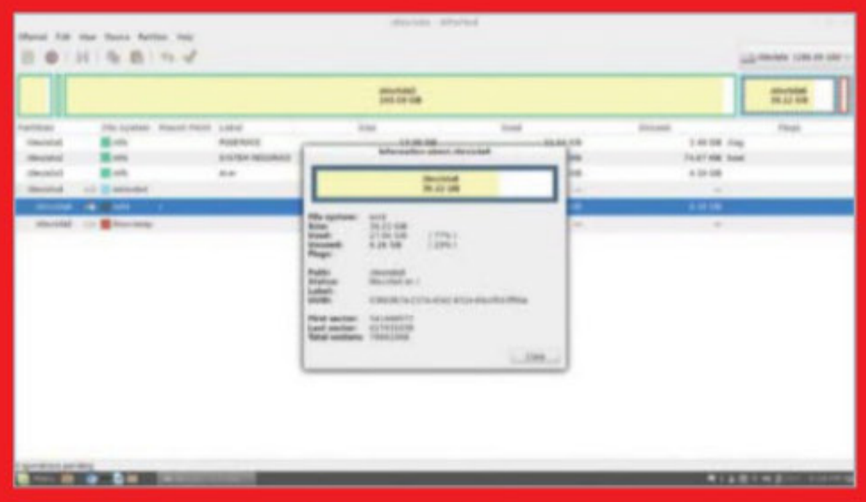
Once you've chosen what to do with the hard drive, Linux Mint will start the install process so you can save some time. The installation itself won't take long, depending on the speed of your system, and it also won't require multiple restarts like when installing Windows.

Editing hard drive space

If you want to have Linux Mint work alongside Windows XP, you'll need to create some space on your hard drive: we always recommend at least 20GB free. This means reclaiming space from Windows, which can be a little dangerous due to file fragmentation.

Basically, Windows likes to split up and spread files around the hard drive while creating, editing or saving them. If you delete space where these files exist you may end up corrupting some files or even break Windows beyond repair. Use the defragmentation software that's included with Windows or UltraDefrag (**ultradefrag.sourceforge.net**) to deal with this first.

Next, you'll need to use GParted, included on the Linux Mint USB or DVD you created. From there you can edit the 'partition' Windows lives on and decrease its size by 20GB. After that you can install safely choosing the option 'Install alongside'.



■ GParted makes the partitioning process easy

Learn your way around

Find out what's what in your brand new operating system with our guided tour

The Computer icon is not quite the same as My Computer

Accessed via the gearbox symbol in the Mint Menu, the Settings menu is similar to the Control Panel but is probably easier to navigate

Working similarly to the Start menu you're familiar with, the Mint Menu has access to apps and the standard shut down options

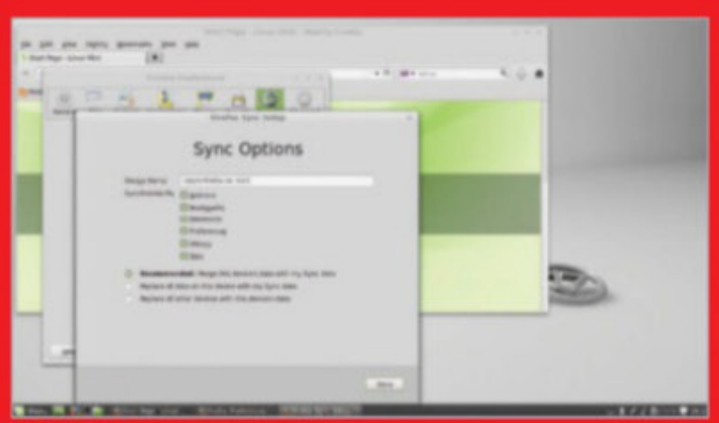
Called Explorer in Windows, this is the file manager for Cinnamon. With drag and drop, double-click, etc, it works in the same way

Found on the side of the Mint menu, you install apps from a choice of thousands via Software Manager – much like on a smartphone

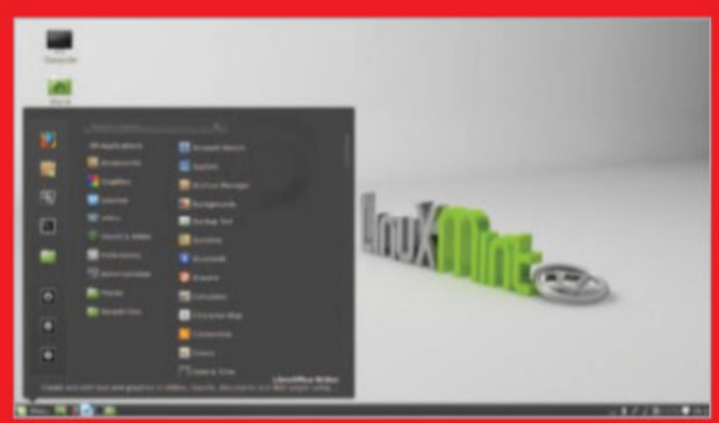
The program bar is where the windows live to switch between. On the right is the standard notifications bar you're used to

Restore your data

01 Restore files
Instead of My Documents, Linux Mint has a Home folder. Here you'll find folders that you can put your files into straight from your USB storage; plug it in and it will pop up on the side of the file window, and you can copy and paste files directly into the Home folder.



02 Restore your app settings
If you used Firefox or Thunderbird before the switch, you can use the profiles you saved or the sync you set up to get your bookmarks, accounts, etc restored with little hassle to the apps already on Linux Mint. Other software like Chrome will also sync if you install it to Mint.



03 Restore quick launch
Linux Mint's Cinnamon desktop will allow you to put quick links to essential apps on the program bar at the bottom. Click the Menu button and search for the app; once you've found what you want, either drag and drop it onto the Quick Launch icons, or right-click and Add to Panel.

Replacement apps

Use these apps that are already installed or available from the Software Manager

Office

One of the most important choices when moving over to Linux is making sure that you know which office suite to use. In our case, LibreOffice is already installed on Linux Mint and is the perfect replacement for Microsoft Office. It's completely compatible with the different file types from all versions of Microsoft Office and can easily be set to save to them as default. It's also regularly updated and a mobile version is on its way very soon, so you can take your files with you.

Alternatives OpenOffice • Calligra Suite

Web browser

Firefox, the open source multi-platform browser, is one of the most popular alternatives to Internet Explorer on all operating systems. Its tabbed approach to browsing was revolutionary at the time and ported into Internet Explorer, and inspired open source rival Google Chrome as well. Linux Mint has this by default and will also accept profile data and synced bookmarks via Firefox Sync, making your migration super easy.

Alternatives Chromium • Midori

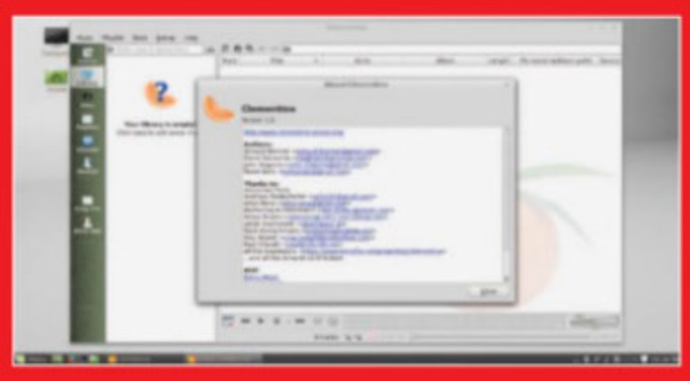
Email client

While Outlook is part of the Microsoft Office bundle, LibreOffice does not have an email equivalent it can use. This is where Mozilla Thunderbird comes in as a sister app to the web browser Firefox; it retains the same functions as Microsoft Outlook but is more customisable and quicker. It integrates well with the Linux Mint desktop, giving you notifications when emails or feed items come in that fit in better with the OS.

Alternatives Claws Mail • Gmail

Music

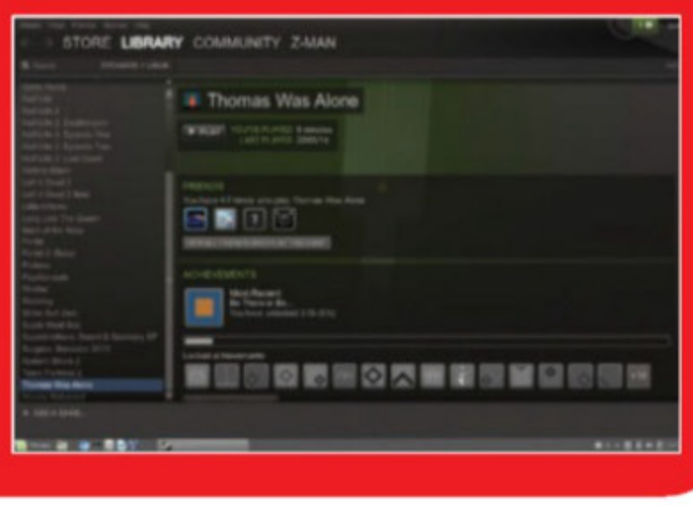
Unfortunately there's no iTunes on Linux, but we'll wax lyrical about Apple another time. What we do have is the excellent Clementine music player: a full-featured music player and media manager that allows you to organise playlists, create smart playlists, play web radio, stream online music and much more. It also hooks into Cinnamon and lets you use any keyboard shortcuts you have for media as well as the notification area for control.



Alternatives Rhythmbox • Audacious

Gaming

Fear not, your expansive Steam account that's been overfilled from summer and winter sales will not go to waste. Steam for Linux is very much a thing and games from the back catalogue are being ported all the time, with newer games releasing on the system as well. It's a major part of SteamOS and Steam Machines, which can be easily installed onto your Linux Mint machine. You'll need to find the Steam repository to install from but once you do you'll get all the updates.



Video

This one is also regularly found on Windows, yet VLC is a lot more at home on Linux. Not requiring any extra media or codec packs, VLC is the best way to watch all of your existing video files and any others that might be invented in the near future. It also works great as a DVD player and won't complain about region encoding on your DVDs either. You'll find it in the Software Manager by searching for VLC.

Alternatives MPlayer • Kaffeine

Settings

Where to go and what to do when you want to tweak the look and feel of Linux Mint

Preferences

Change how Linux Mint and the desktop works, such as changing default applications with Preferred Applications or adding extra virtual desktops to switch between so you can keep your computer screen more tidy.

Hardware

The two important options here are Device Drivers and Software Sources. Device Drivers allows you to install proprietary drivers for your PC if there is no open source equivalent. Software Sources allows you to add more software repos if you want to install apps not originally included with Mint.

Administration

Here we have settings directly linked with the hardware, such as fine-tuning the mouse with Mouse and Trackpad, adding and managing printers, setting up networks and configuring Bluetooth if it's available.

Linux User & Developer

SUBSCRIPTION VOUCHER

☑ YES I would like to subscribe to Linux User & Developer

YOUR DETAILS

Title_____ First name_____

Surname_____

Address_____

_____

_____

_____

Postcode_____ Country_____

Telephone number_____

Mobile number_____

Email address_____

Please complete your email address to receive news and special offers from us

DIRECT DEBIT PAYMENT

☐ **UK Direct Debit Payment**

I will pay only £18 for 6 issues (saving 50%*)

Instruction to your Bank or Building Society to pay by Direct Debit

DIRECT Debit

ip IMAGINE PUBLISHING

Please fill in the form and send it to: Linux User and Developer Subscriptions Dept, 800 Guillat Avenue, Kent Science Park, Sittingbourne ME9 8GU

Name and full postal address of your Bank or Building Society

To: The Manager — Bank/Building Society

Address

Postcode

Originator's Identification Number: 5 0 1 8 8 4

Reference Number

Instructions to your Bank or Building Society
Please pay Imagine Publishing Limited Direct Debits from the account detailed in this instruction subject to the safeguards assured by the Direct Debit guarantee. I understand that this instruction may remain with Imagine Publishing Limited and, if so, details will be passed on electronically to my Bank/Building Society

Name(s) of account holder(s)

Branch sort code

Bank/Building Society account number

Signature(s)

Date

Banks and Building Societies may not accept Direct Debit instructions for some types of account — A6 instruction form

PAYMENT DETAILS

YOUR EXCLUSIVE READER PRICE, ONE YEAR (13 ISSUES)

☐ **UK: £62.40 (Save 20%)** ☐ **Europe: £70.00**

☐ **World: £80.00**

Cheque

☐ **I enclose a cheque for £**_____

(made payable to Imagine Publishing Ltd)

Credit/Debit Card

☐ Visa ☐ MasterCard ☐ Amex ☐ Maestro

Card number _____ Expiry date _____

Issue number ☐☐ (if Maestro)

Signed _____

Date _____

Code: PAL142Q

Please tick if you do not wish to receive any promotional material from Imagine Publishing Ltd
by post ☐ by telephone ☐ via email ☐

Please tick if you do not wish to receive any promotional material from other companies
by post ☐ by telephone ☐ Please tick if you DO wish to receive such information via email ☐

Return this order form to:

Linux User & Developer Subscriptions Department, 800 Guillat Avenue, Kent Science Park, Sittingbourne, ME9 8GU or email it to lud@servicehelpline.co.uk

You will be able to view your subscription details online at www.imaginesubs.co.uk

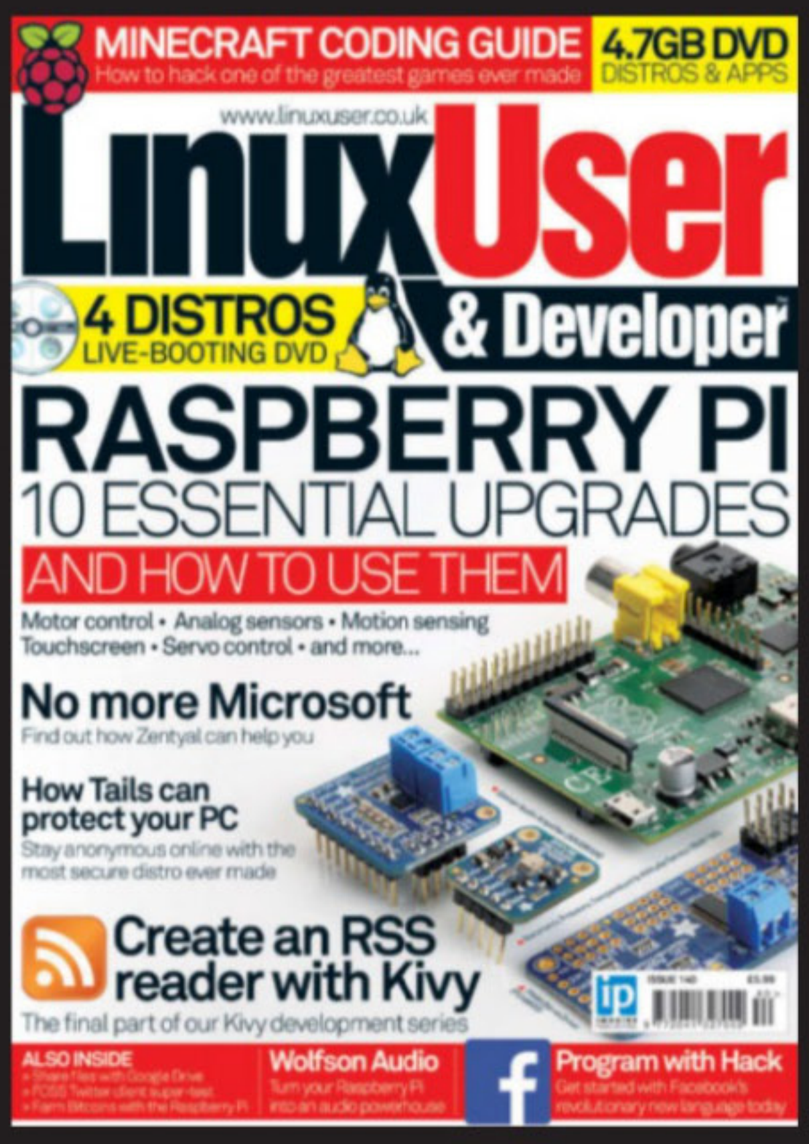
FANTASTIC SUBSCRIBER OFFER

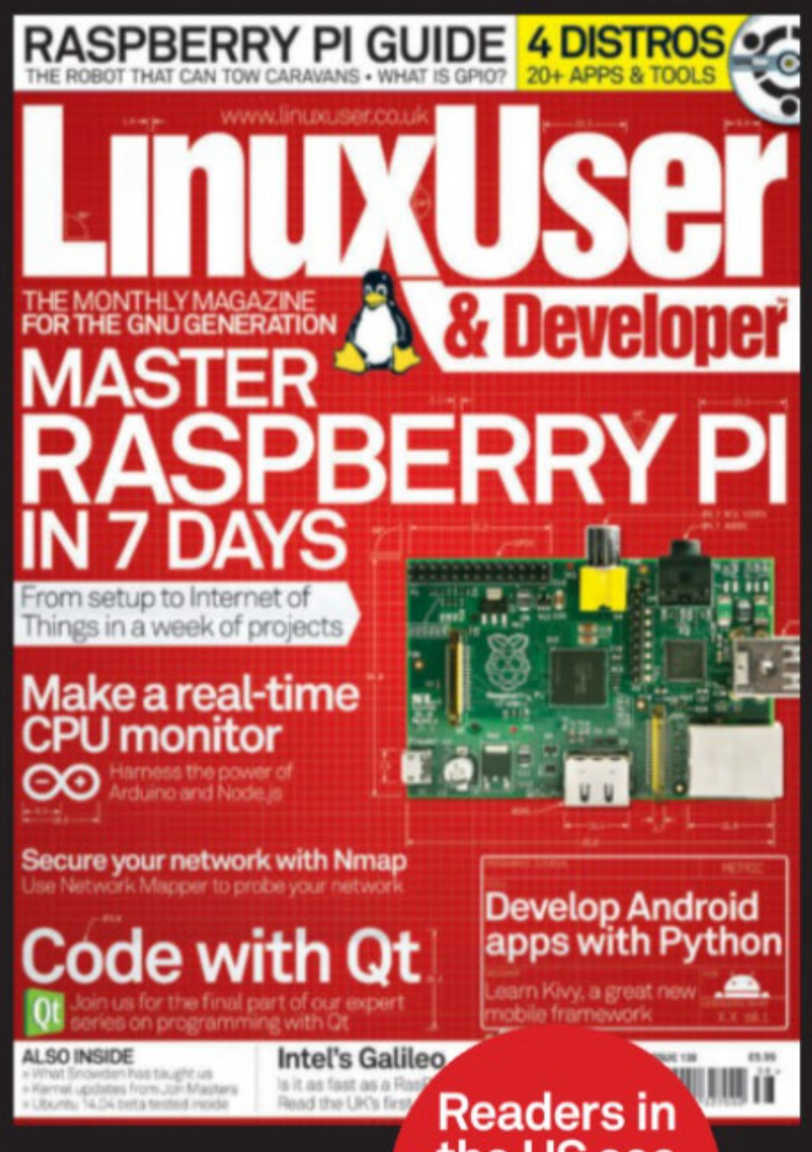
Subscribe today and save 50%

- Pay only £3 for every future issue – a 50%* saving on the store price
- Free UK delivery to your door
- Never miss an issue
- Money-back guarantee

*Terms & conditions: Pricing will revert to our standard offer of £25.15 every 6 issues on the third payment made. Subscribers can cancel this subscription at any time. New subscriptions will start from the next available issue. Offer code PAL142Q must be quoted to receive this special subscription price. Details of the direct debit guarantee are available on request. Offer expires 31st October 2014. Imagine Publishing reserves the right to limit this type of offer to one per household.

SUBSCRIBE TODAY & SAVE 50%*





Readers in the US see page 84 for exclusive offer

THREE EASY WAYS TO SUBSCRIBE

1. Online

www.imaginesubs.co.uk/lud

And enter PAL142Q

2. Telephone

0844 249 0282

And quote PAL142Q

3. Post or email

Please complete and post the form to

Linux User & Developer
Subscriptions Department
800 Guillat Avenue
Kent Science Park
Sittingbourne
ME9 8GU

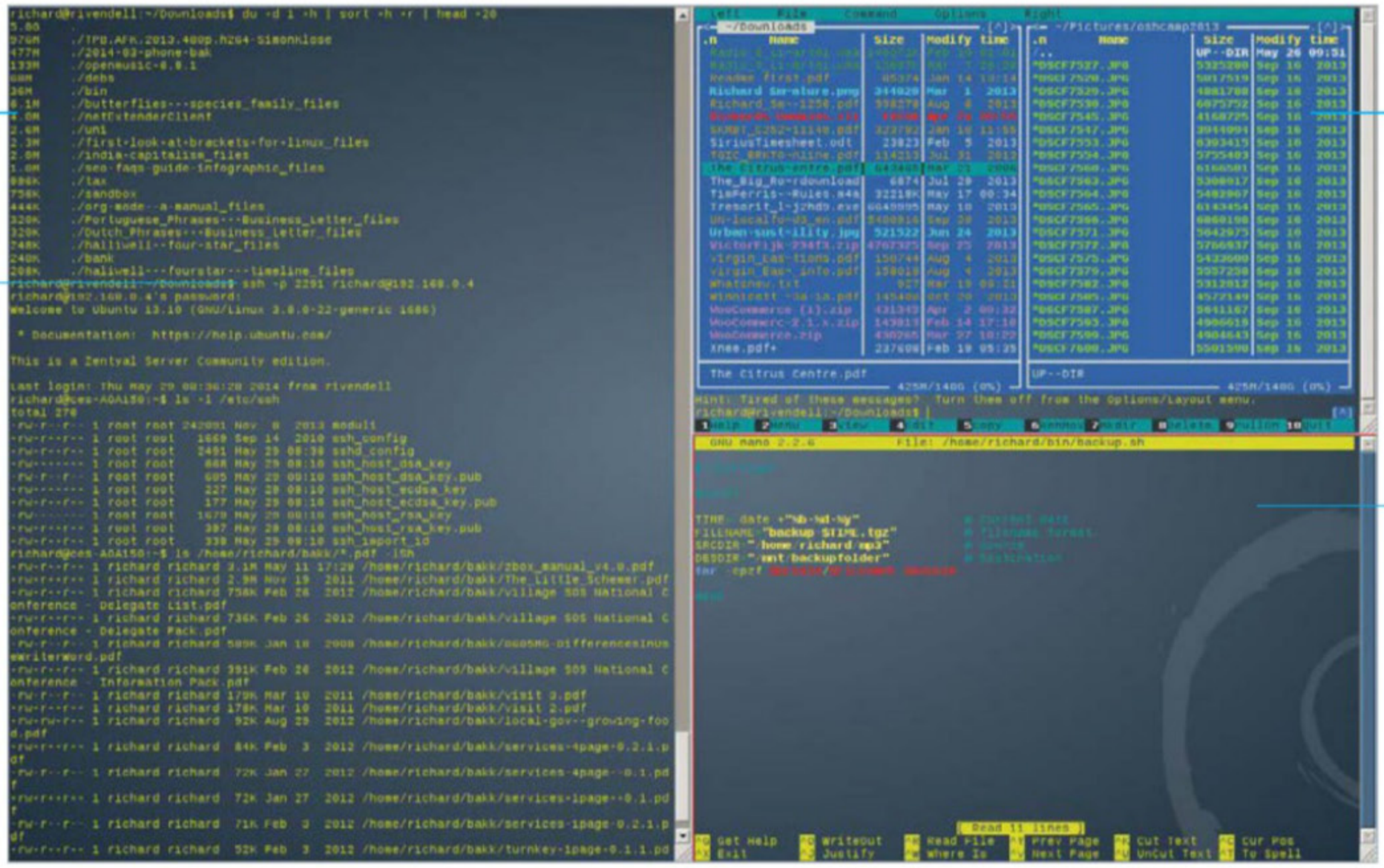
Or alternatively, you can scan and email the form to

lud@servicehelpline.co.uk

Spring clean your files using the command line

Use the Unix pipe to chain commands together - the output from the first process becomes the input for the next

MC - Midnight Commander - is as close to a GUI as you can get inside a terminal



A remote SSH server makes it easy to receive automated backups from your scripts, or to send copies to your file server from anywhere

Putting the commands in a shell script saves you remembering all the steps and allows automation and scheduled execution

Spring clean your files using the command line

Sort through your hard drive, find lost files and archive old stuff far quicker than you could through your GUI file manager

Advisor

Richard Smedley started using computers long before WYSIWYG, and still maintains that the command-line, and Emacs, is the most productive working environment

Resources

/bin/bash

So computing is clouding over, but you have half a lifetime's 'stuff' randomly saved across hard drives and thumb drives. Family photos; downloaded eBooks; text files helpfully named 'Notes' and 'Misc'; PDFs of instruction manuals, brochures and cartoons; HOWTOs in HTML form; music and video files; lots of old source code; and even files you don't even know how to open.

Spring has come and gone and still your disordered archive sits quietly but accusingly. It's time to do something about it – but drag-and-drop sorting through everything will take ages. Instead, let's brush up on some command line shortcuts and combine a tidying exercise with a learning opportunity.

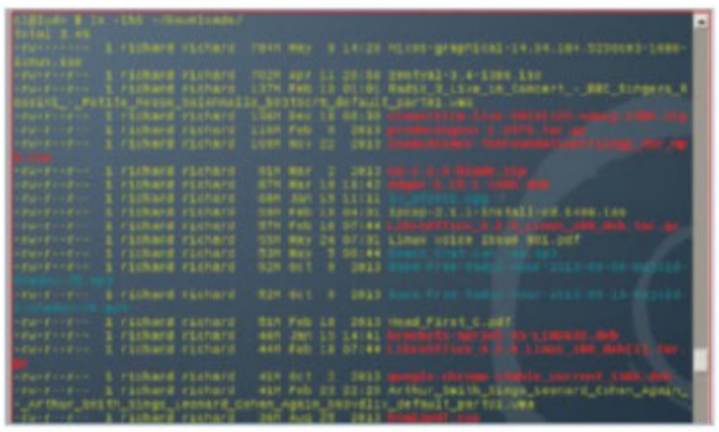
Why do this from the command line? There are times when drag-and-drop is a convenient way to move files around but - for most operations involving several files in multiple directories - the command line can be a lot quicker. Not only that, but repetitive operations can be automated with scripts and scheduled jobs, as we'll see later.

Read on and learn how to find files, automatically zip them up, collect them in backup partitions and send them securely to a remote server at scheduled intervals.

Spring clean your files using the command line

Archive your old stuff quicker than you could through your GUI file manager

TUTORIAL



01 Big list

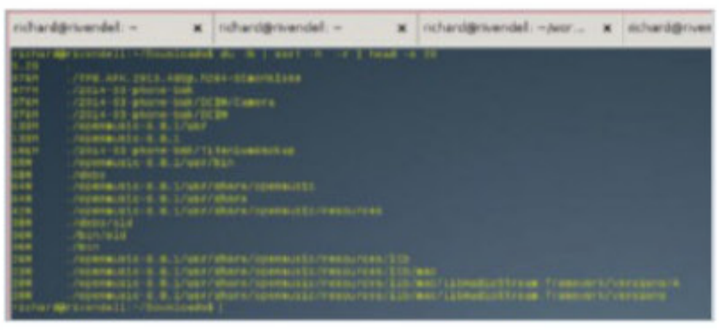
There are many ways to see how big a file is. Right-click>Properties in a GUI file manager or **ls -l** filename, but **ls -lhS** (**h** to give human readable units) gives you a good look at the whole directory, ordered by file size.

02 Du it

But if you're just looking for disk use – not every file attribute – the disk use program, **du**, will be your best friend. Best of all, it searches recursively within directories.

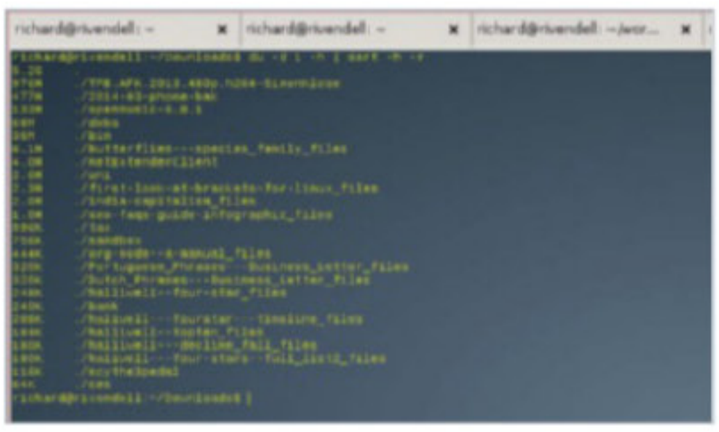
03 Pipe up

One of the great powers of *nix shells is piping commands together with the | pipe. This takes the output of one program and feeds it as the input to the command after the |.



04 Sort it out

For example, feeding output to **sort** – in particular **sort -n -r**, which puts a list in numerical order (**-n**), from largest to smallest. **-h** tells **sort** to take into account KiB, MiB, etc, rather than **-n**'s ignorance of units.



05 In depth

du's recursive nature, while handy for discovering what's filling your subdirectories, is a pain when you're looking for a less detailed overview. Setting depth to 1 is some help...

```
richard@rivendell:~/Downloads$ find . -type f -printf "%k %p\n" | sort -rn | head
998392 ./TPB.AFK.2013.480p.h264-SimonKlose/TPB.AFK.2013.480p.h264-SimonKlose.mp4
803004 ./nixos-graphical-14.04.164.5250ce3-i686-linux.iso
718624 ./zentyal-3.4-i386.iso
140420 ./Radio_3_Live_in_Concert_-_BBC_Singers_Rossini_-_Petite_Messe_Solenn
5_default_part01.wma
138692 ./clonezilla-live-20131125-saucy-i386.zip
118792 ./producingoss-1.2479.tar.gz
111204 ./IsaacAsimov-TheFoundationTrilogy_vbr_mp3.zip
92460 ./cm-7.2.0-blade.zip
88696 ./edgar-1.15-1.i686.deb
70420 ./openmusic-6.8.1/usr/bin/openmusic
richard@rivendell:~/Downloads$
```

07 Helpful find

> **For operations involving files in multiple directories the command line can be quicker**

06 In the loop

While this combination is in part a workaround for non-GNU versions of sorts, on some versions of *nix, it also has the bonus advantage of showing large individual files and introduces loops in the shell.

07 Helpful find

A better look, if you're concerned with files, not directories, may be achieved through properties of the helpful **find** command. Here we're using **-type f** to limit the search to files, and **-printf %k %p** to show size (in KiB) and filename.

08 Space hogs

Having identified the type of files that are taking up the most space, in order to hunt them across your disk drive you will need to use the case-insensitive **-iname** switch and apply find to the root directory.

09 Take your time

Searching across the whole filesystem takes real time – eight minutes here – and throws up the odd /proc irrelevance, but it is thorough and may surprise you with disk hogs you'd forgotten about, like old mailboxes.

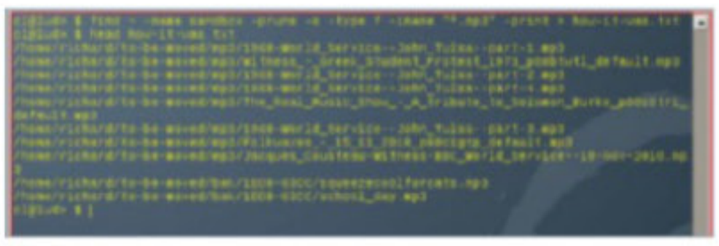
Spring clean your files using the command line

10 Music and movement

We're going to use **find** to locate all of the MP3 files on the system – or at least those within the home directory – and move them to another disk. But before we do this we need to know what not to move...

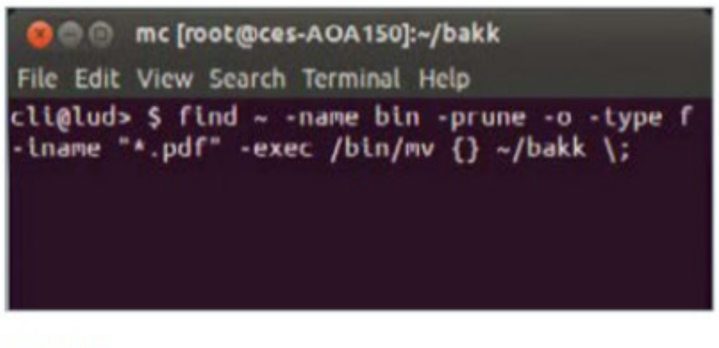
11 Exclusions

You may have games, or perhaps system sounds, in MP3 format, so – having searched the **find** results to double-check on them all – tell **find** to ignore the directories where they can be found.



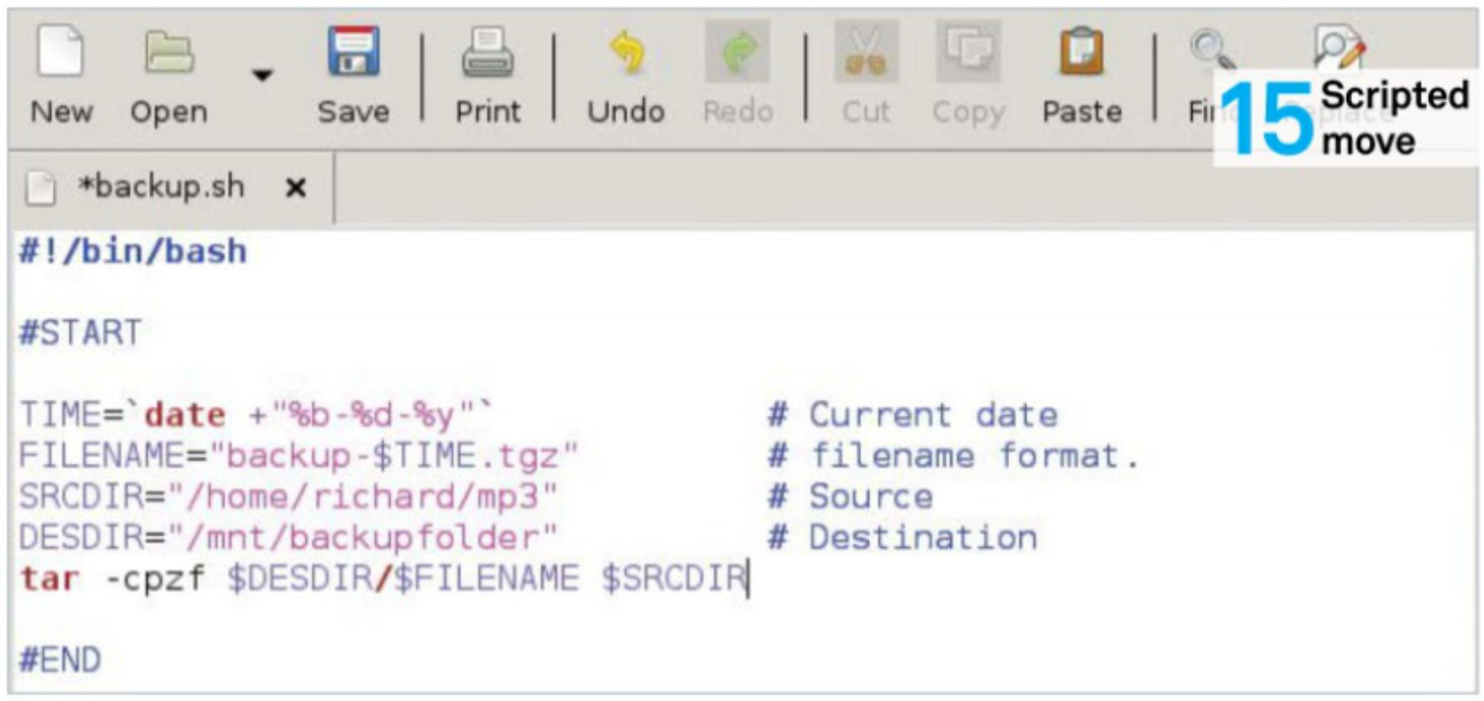
12 Snapshot

You're going to be moving, not copying – this is a space-saving exercise – so take a snapshot of where the files are before you make any changes. That way you'll know where to put things back, if necessary!



13 Executive step

Calling **-exec** allows a command – in this case **/bin/mv**, the move command – to execute on each of the outputs of **find**. This is the safest option, as it takes care of special characters, and works on most *nix variants.

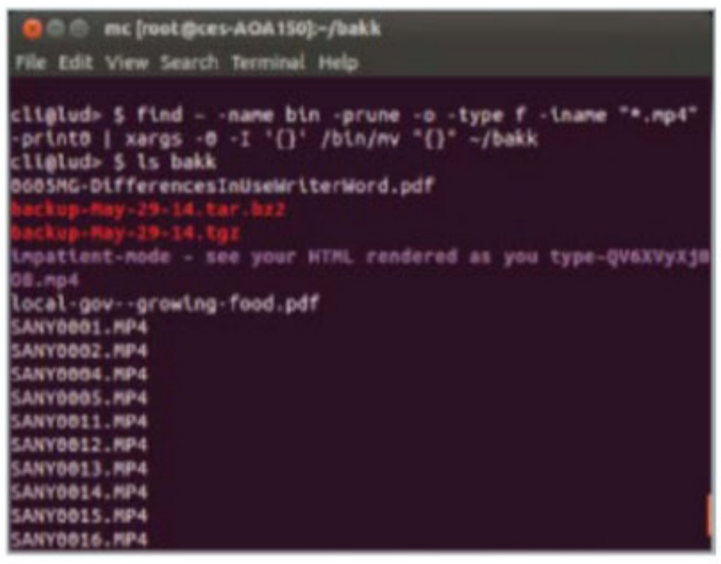


15 Scripted move

If the object of your spring clean is just to tidy similar files into one directory, the job is done – but not quite. The time to create backups is always right now, since you don't know when your disk will fail, or if your laptop will be lost or stolen.

A short script will take those files you've collected together, zip them up into a tarball, compress it and put it on another mounted directory. You might rather move your backup to a remote file server, for example. We'll examine the options over the next few steps.

> **An ARM device would be a good option for backups**



14 Xargs

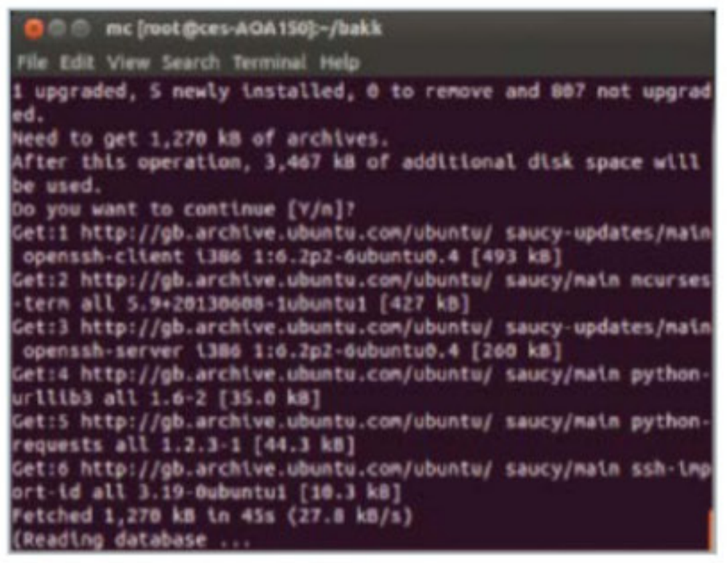
xargs is faster – running over the entire output of **find** at once – but needs **print0** (not on all *nix) to safely operate special characters. Note these last two commands are each on just one line, don't hit Enter till the end.

16 Arm box

Backups need somewhere convenient to go. If you don't already have a media box or other file server, but want to set something up, we recommend using something that's quiet, unobtrusive and has low-power consumption: an ARM device – whether an old NSLU-2, a firmware-modified router or the good old Pi – would all be good options here.

17 More Pi please

The real decision is software. Keep it simple: Debian. Forget NFS, SAMBA, or other remote file system mounts...



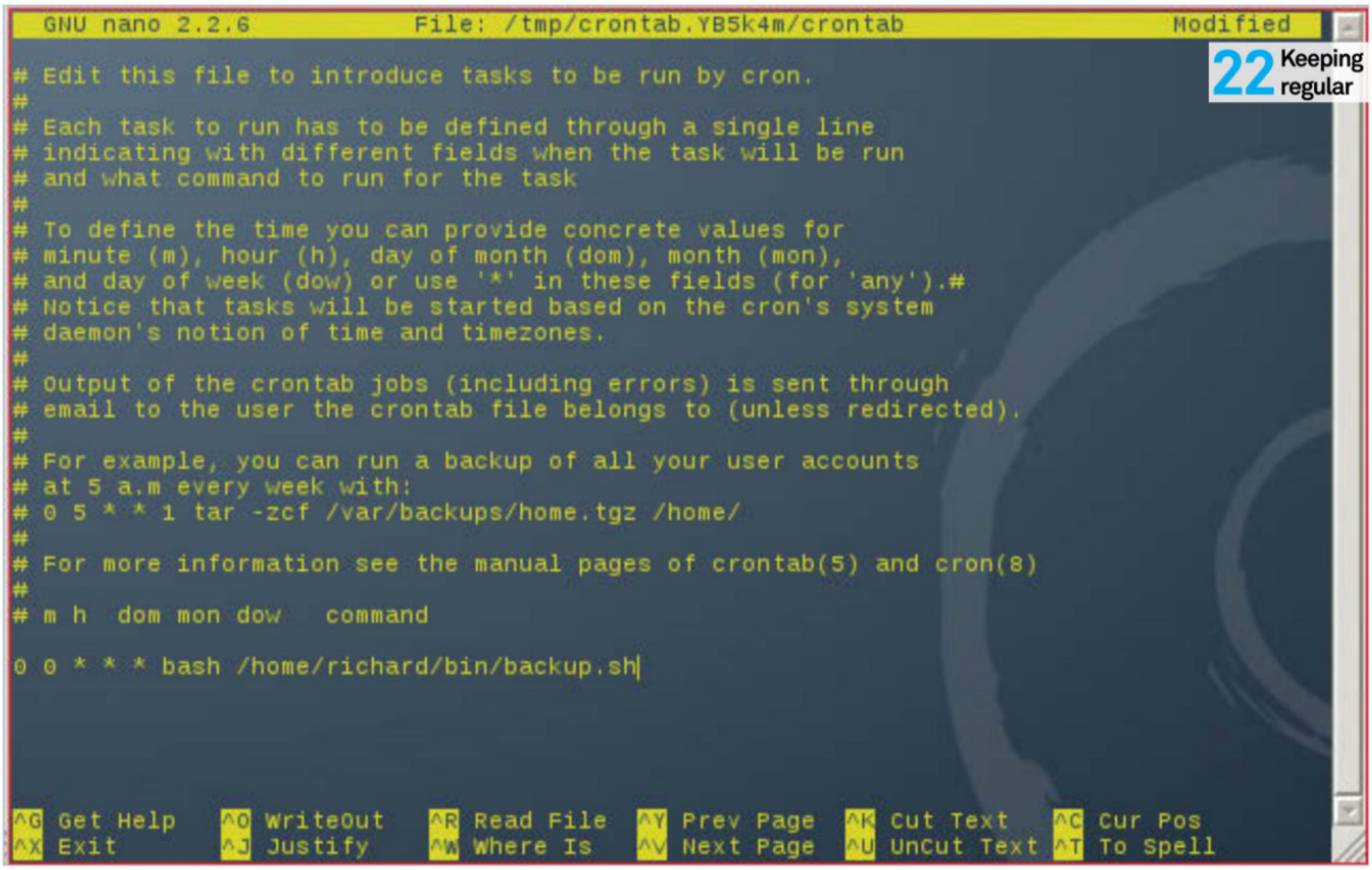
18 Secure backup server

... just install the OpenSSH daemon. Now you can use SCP – secure copy – to backup your files in your scripts. Here we're using an old Ubuntu netbook to upcycle as a media and print server – and a programmable photo frame!

Spring clean your files using the command line

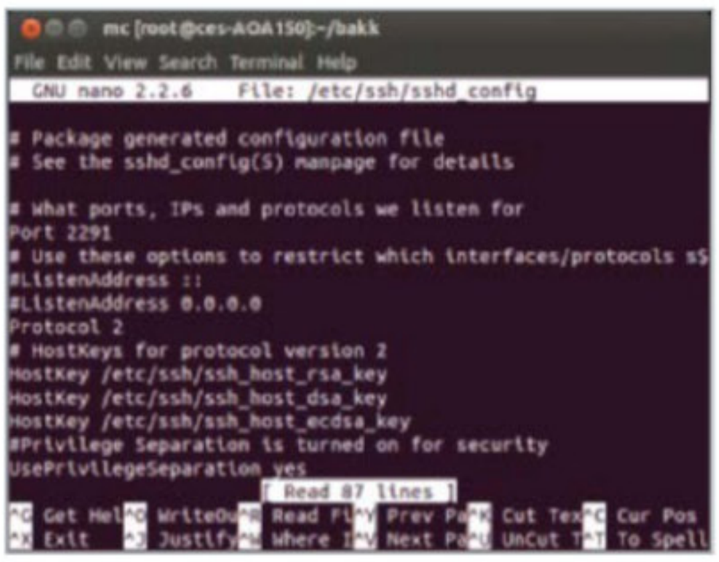
Archive your old stuff quicker than you could through your GUI file manager

TUTORIAL



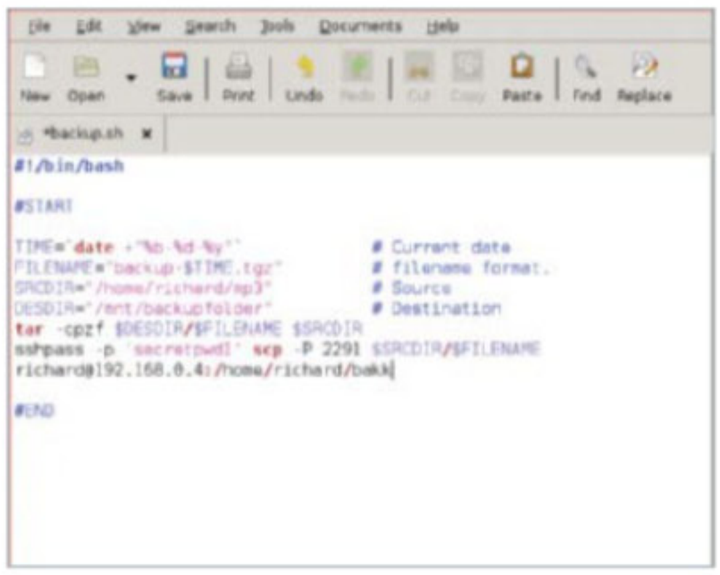
19 Key or pwd?

Your SSH server has a public key, but adding one from your local machine (to match its private key) would enable you to automate logins without having to leave the SSH password in plain text in your scripts.



20 Obscure your port

While we'll leave SSH keys as an exercise for you (though we may return to it in a future issue), we do suggest opening /etc/ssh/sshd_config on your backup server and changing the value of **Port** to a random high number.

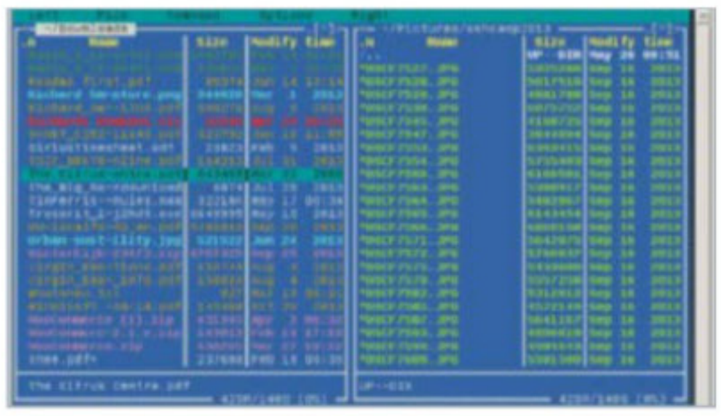


21 Secure copy

If you didn't set up keys, then install Sshpass – it's the best way to send a password to SCP in a script – and add the line above into your script. It's all on one line.

22 Keeping regular

Now you have a script, you can call it through **cron** to run regularly with **crontab -e**. For example, if you generate and/or download many PDFs in a working day, or have particular folders to save, put those in the script.



23 Command line GUI

Before we go, having shown you the power of the command line, we're not saying GUIs don't have a place. Install MC and you'll get all of the power of keyboard-driven commands in a two-pane terminal-based file manager.

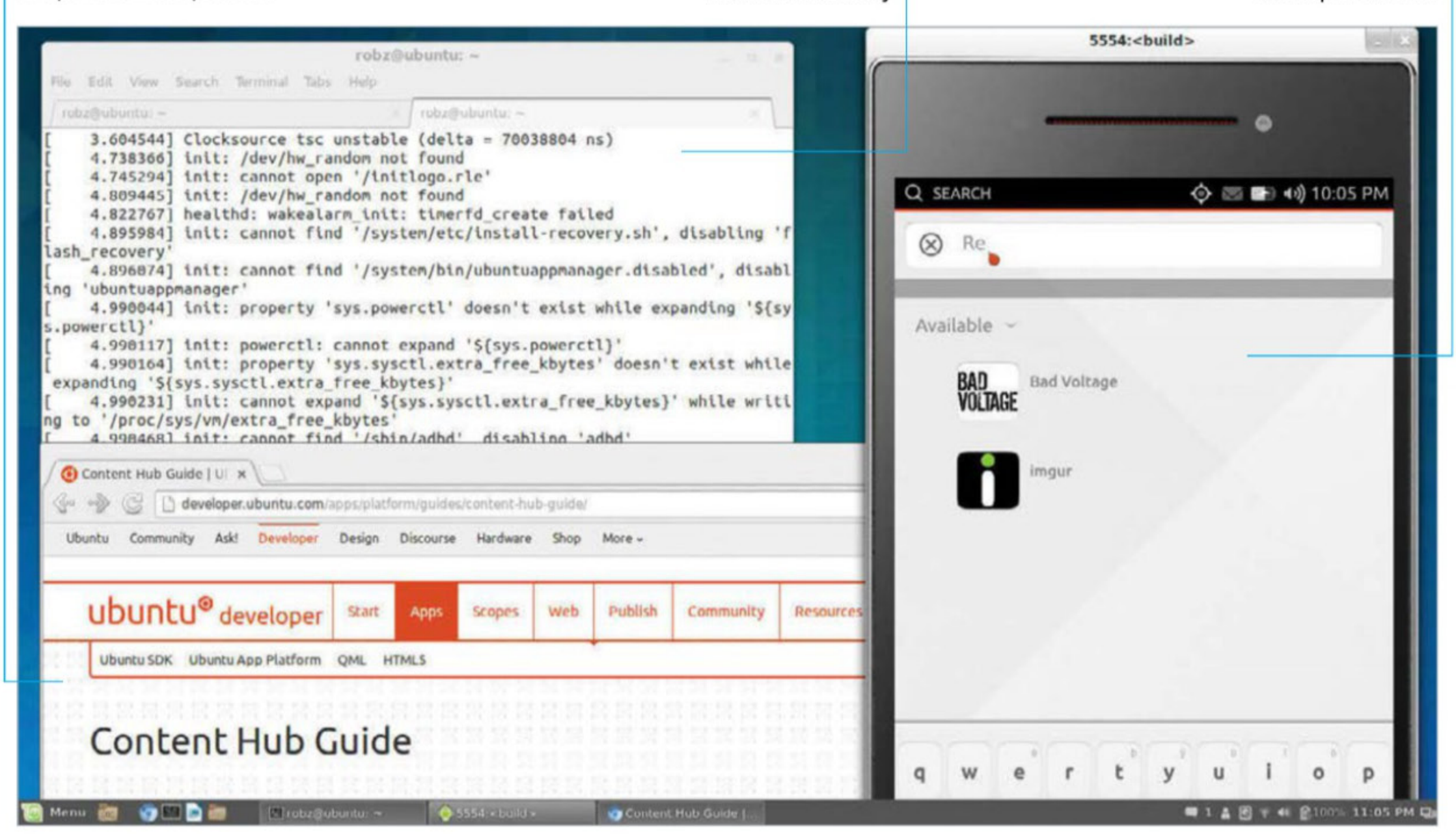
24 More scripts

In a future issue we'll go further with shell scripting: the chaining together of programs and built-in commands in short scripts to remove the repetition of running through the same operations time and again. Happy days!

Get access to more by using the Ubuntu SDK to develop apps for your emulator, creating and managing snapshots in the process

Connect to the emulator using SSH and ADB to transfer files and send commands, and to control the instance remotely

Emulate a fully working Ubuntu Touch image to check out the mobile OS or begin development for it



Emulate Ubuntu Touch

Advisor

Rob Zwetsloot models complex systems and is a web developer proficient in Python, Django and PHP. He loves to experiment with computing

Aid development for and generally test out Canonical's own phone and tablet OS using the Ubuntu Touch emulator

Within the next year we should be getting our first glimpse of Ubuntu phones, powered by Canonical's own mobile OS: Ubuntu Touch. It's been available for use on various Nexus devices for just over a year now, however there is an alternative method of using it in the Ubuntu Touch emulator.

The emulator allows you to step into the Ubuntu Touch interface with all the relevant controls you could possibily need in order to properly emulate a phone or tablet. As well as finding out how the interface generally works, it's an excellent platform for testing out Ubuntu Touch apps and developing for it.

The emulator does not emulate the ARM architecture though, and as such not all the current apps work on it. In the future most, if not all, apps will be able to work across all platforms, including the standard desktop version of the distro.

So, if you don't want to risk bricking your phone, let's get started!

Resources

An Ubuntu install
www.ubuntu.com/download/desktop

Ubuntu Touch emulator
https://launchpad.net/phablet-tools

Emulate Ubuntu Touch

Aid development and test out Canonical's own phone and tablet OS

TUTORIAL

01 Get the tools repository

If you're using Ubuntu 14.04 or later you'll already have the phablet tools repo installed. 14.04 users can skip this step, but for everyone else you'll need to add the repository so you can install the relevant software. Do this with:

```
$ sudo apt-add-repository ppa:phablet-team/tools
```

02 Install the emulator

Press Enter to confirm the PPA and then run an **apt-get** update for it to populate APT with the new available software. Now you can start the process of installing Ubuntu Touch to a mobile device – but for the emulator you'll need to do:

```
$ sudo apt-get install ubuntu-emulator ubuntu-emulator-runtime
```

03 Create your instance

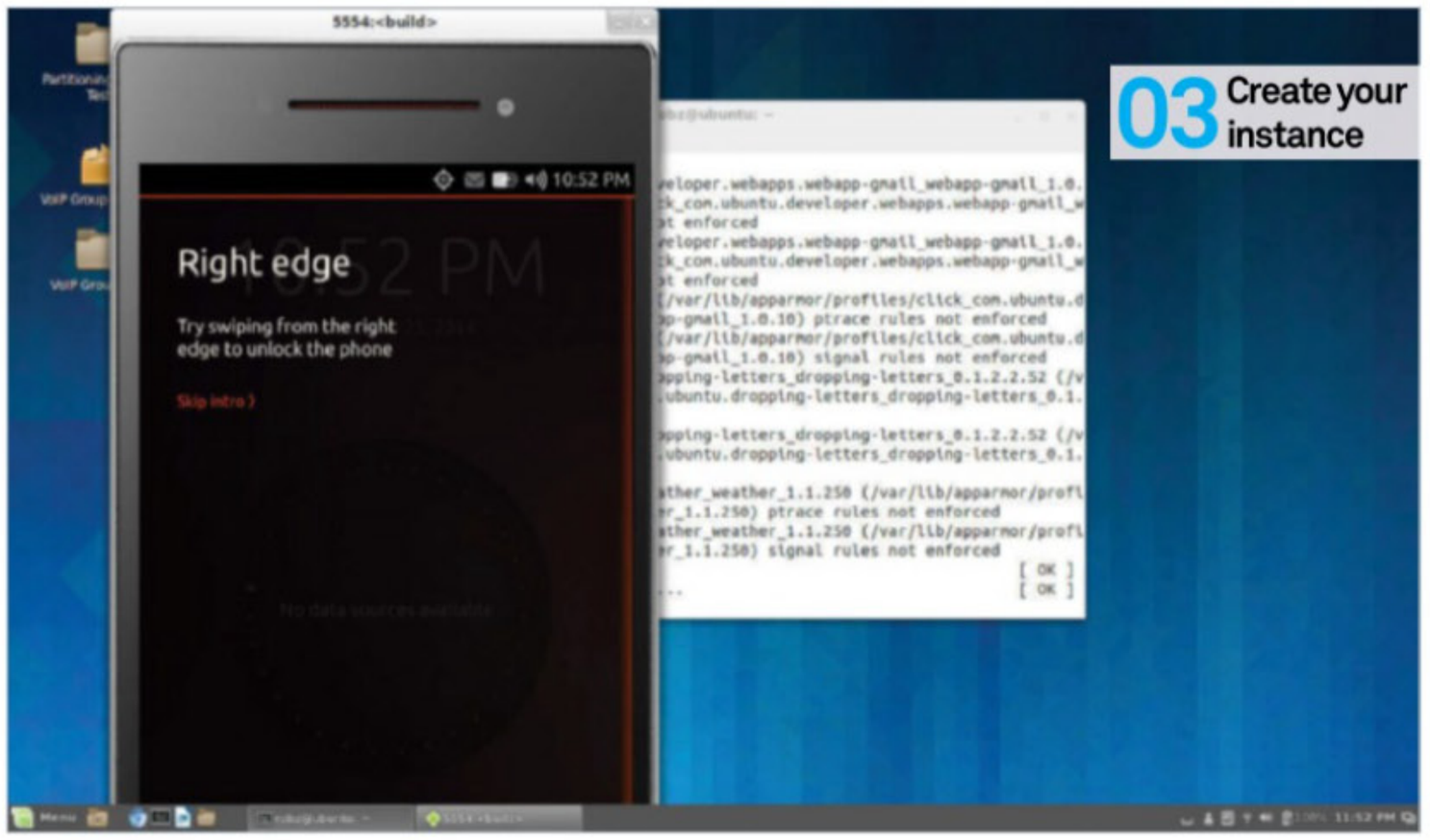
To start your emulated phone you need to first create an instance by downloading the current image and then running it. Name it whatever you wish but remember its name for the future:

```
$ sudo ubuntu-emulator create [instancename]
$ ubuntu-emulator run [instancename]
```

04 Change the architecture

If you're having issues launching the emulator after building it like this, you can also create an instance with your specific CPU architecture in mind. The safest thing to do on both x86 and x86_64 is to do this:

```
$ sudo ubuntu-emulator create --arch=i386 [instancename]
```



05 64 problems

On 64-bit machines, if you're still getting issues then the solution for now is to install the i386 dev library. It's a known bug so future versions will fix it, but for now you can install it with:

```
$ sudo apt-get install libudev1:i386
```

06 Using the phone

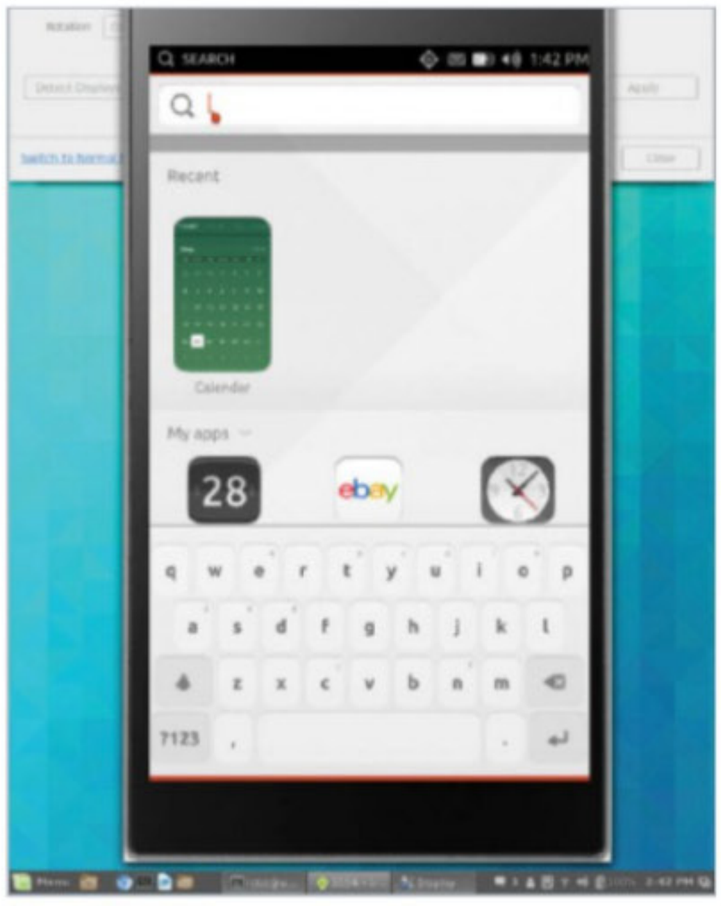
The main way to use your emulated Ubuntu Touch device is by clicking the mouse and moving it to swipe. There's a little tutorial to help you along the way, however the basics involve swiping down from the top to get settings, swiping from the left for apps and moving between frames by swiping from the right.

07 More usage tips

A click is considered a tap, so use it to tap on apps and further move around the interface. The keyboard is not hooked in so you'll need to manually type by tapping on the letters with your mouse in order to input text.

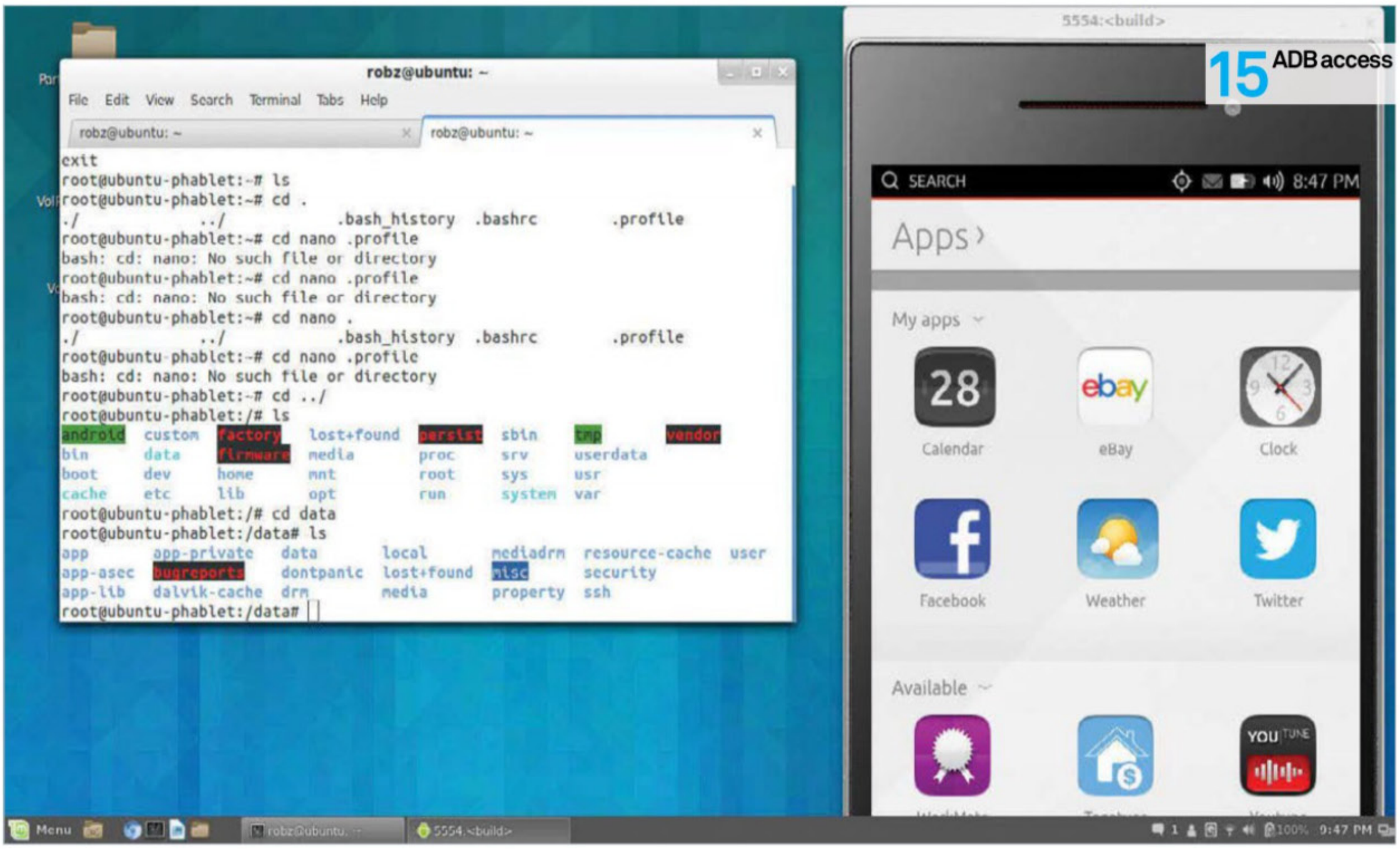
08 Orientation

If you're working on a laptop or don't have a massive screen, you may not be able to see the entire emulated instance. One way to get around this is to change the orientation of the display to portrait; this can be done in the display settings of most desktop environments. Alternatively, press Ctrl+F11.



09 Exiting

The best way to end the instance is to use the window's Close button. This will properly save any settings you've made and anything you've changed to make sure you can turn it back on to its previous state. Trying to kill the script from the terminal will not terminate the instance properly, forcing a restart if you want to be able to use it again.



> **ADB works better in editing and updating your phone content**

10 Use ADB

The Ubuntu SDK and Touch emulator is built upon Android and ADB technology, so you can control it using the same commands as you would an Android device. You can find a list of devices and emulators attached with:

```
$ adb devices
```

11 SSH in

While the emulator has SSH installed, it's not activated by default. You can enter the emulator by using ADB Shell in the terminal. You will then need to access the root account by using:

```
$ su phablet
```

12 Start SSH

Once in root we're going to start the SSH service and then set it so it can autostart after a reboot. We'll do this with the following commands:

```
# service ssh start
# setprop persist.service.ssh true
# sudo reboot
```

The password is phablet.

13 SSH routing

Finally, you'll need to do some clever routing of the SSH ports to do it via localhost. The emulator software handles most of it, so just do it by entering the below into the terminal:

```
$ adb forward tcp:9999 tcp:22
```

14 SSH access

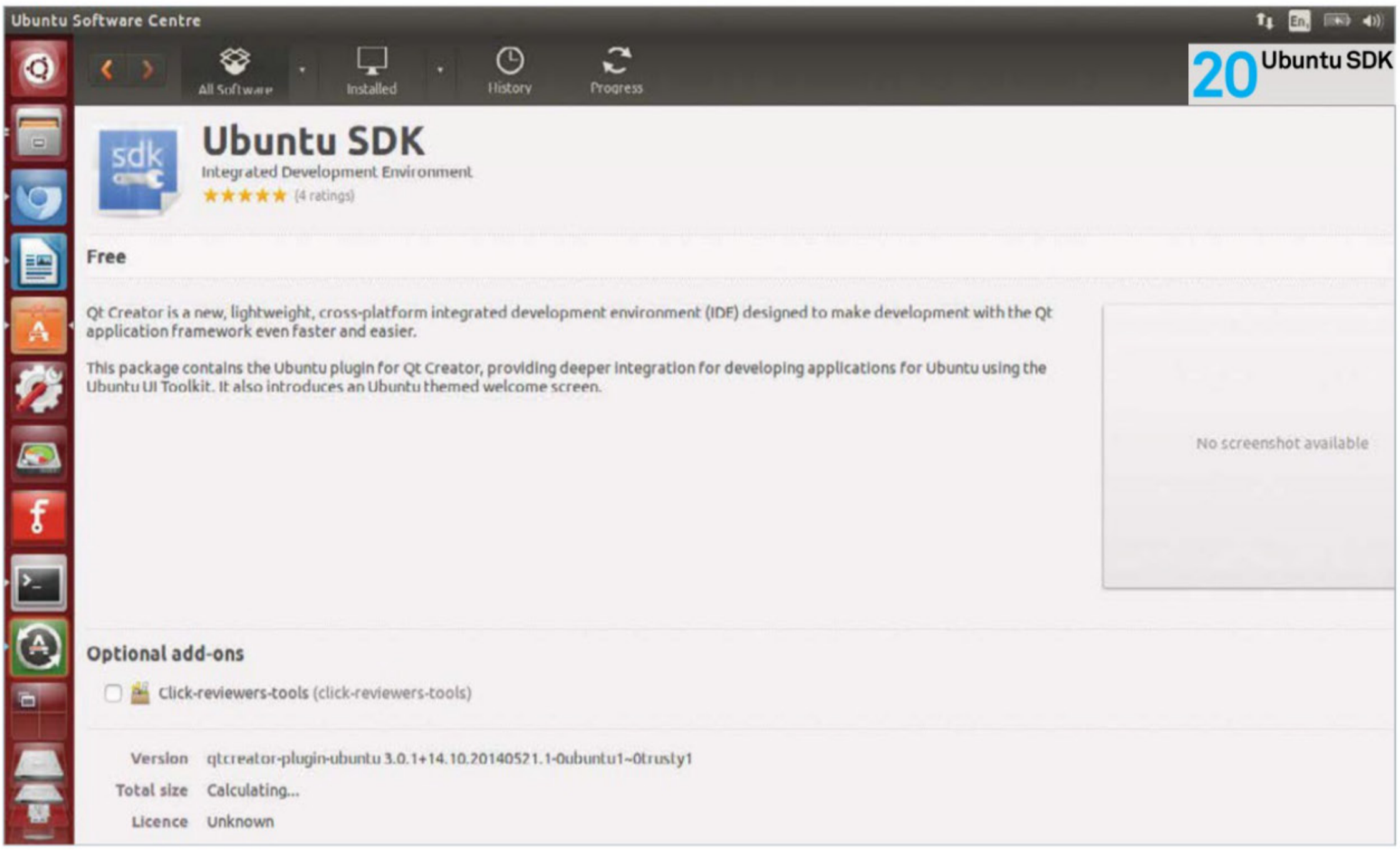
You can now enter into Ubuntu Touch with SSH by using the **ssh** command. You'll likely have to enter the password again, and you'll also need to confirm the SSH credentials of the emulator:

```
$ ssh phablet@localhost -p 9999
```

Emulate Ubuntu Touch

Aid development and test out Canonical's own phone and tablet OS

TUTORIAL



15 ADB access

SSH isn't the only way to access the emulated image: using ADB as we did earlier can be much quicker, and ADB also has access to specific functions that allow it to work better in editing and updating phone content. As before, you can enter directly into the emulator just by using:

```
$ adb shell
```

You are then able to navigate the file system in the same way you would with **cd**, run commands and install software. As before, the password for the whole thing is phablet if you want or need to run as root.

16 Take a snapshot

You can create a snapshot of your emulated image to revert to when necessary if changes are to be made. To do this, make sure you first remember the name of it, close down the running image and enter:

```
$ ubuntu-emulator snapshot
--create=pristine [instancename]
```

17 Reverting snapshots

These snapshots can be useful for reverting back to earlier versions for development purposes. There are two ways of doing it: pristine and normal. Pristine revert goes back to how the snapshot looked when it was first created. This can be achieved like so:

```
$ ubuntu-emulator snapshot --revert-
pristine [instancename]
```

18 Back to normal

If you want to revert to a later, normal version of a screenshot with extra settings and features that you've set up, you can do this with:

```
$ ubuntu-emulator snapshot
--revert=pristine [instancename]
```

19 Extra key commands

As well as being able to use Ctrl+F11 to change orientation, there are some extra keyboard functions you can use to access hardware keys. These include using F7 instead of the power button and Ctrl+F5 and Ctrl+F6 to decrease and increase volume respectively.

20 Ubuntu SDK

The Ubuntu SDK is used for developing apps on Ubuntu Touch and also includes its own way of connecting to and creating emulators to run your code. The SDK itself can be installed via the following:

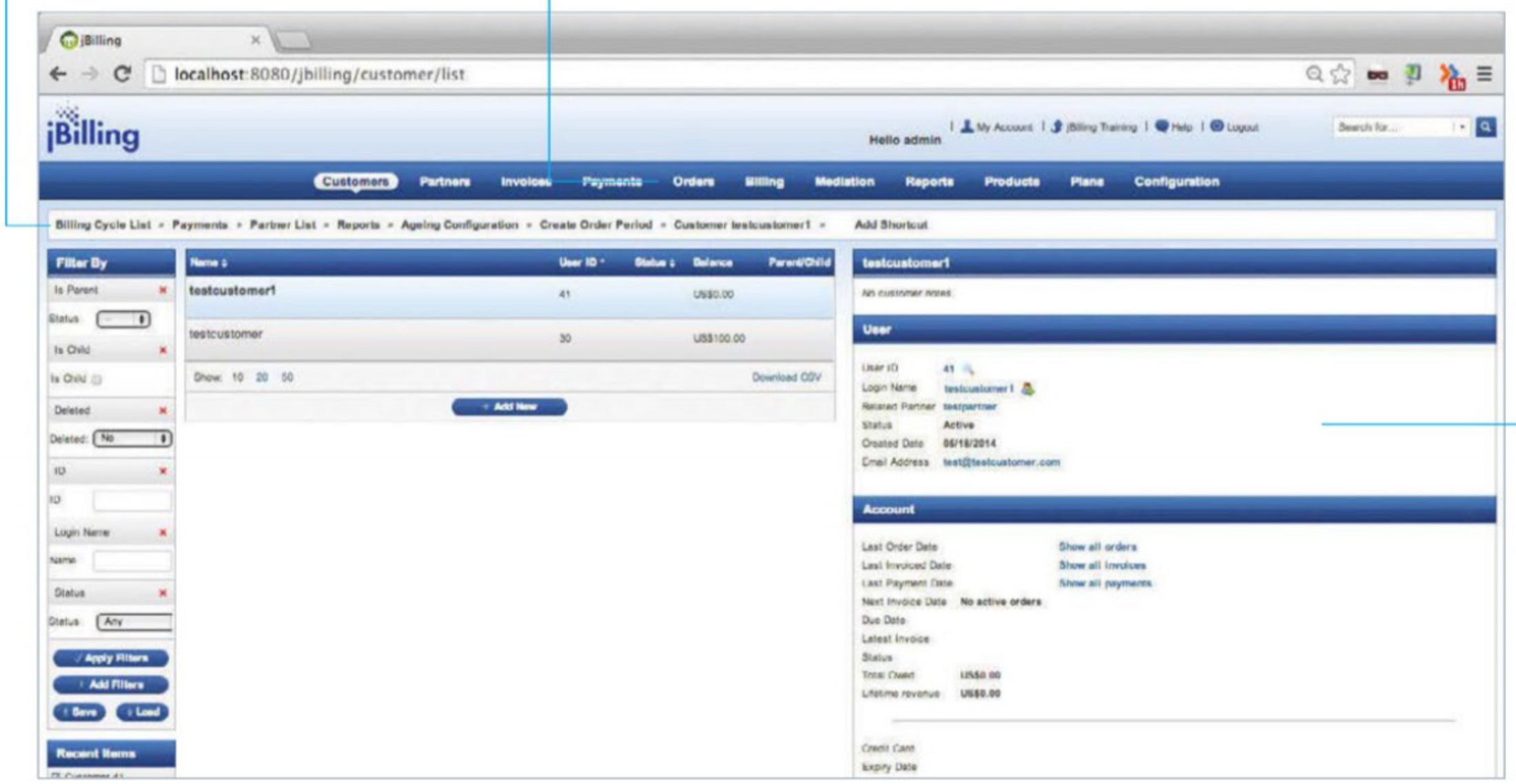
```
$ sudo apt-get install ubuntu-sdk
```

21 SDK emulator

When the emulator is running you can attach it to the SDK to test your apps and such. To do this, open the Devices tab and go to the Device Actions subtag within that. Click on Redetect devices and your emulator should be added to the Serial number field to select.

22 Keep testing

With the emulator in place and working, you now have full control over what you want to do with it! Testing out the OS as well as testing out apps is a good way to get into Ubuntu Touch for when the phones start hitting the market.



Build a billing platform for your business with jBilling

Discover jBilling and make managing invoices, payments and billing simple and stress-free

Advisor

Nitish Tiwari is a software developer by profession and an open source enthusiast by heart. As well as writing for leading open source magazines, he helps firms set up and use open source software for their business needs

Resources

jBilling www.jbilling.com/community

A lot more people are taking up the entrepreneurial route these days. To the uninitiated it looks very easy; you are your own boss and can do whatever you wish. But someone who has already taken the plunge knows that being an entrepreneur is a lot tougher – whether working as a freelancer or the founder of a start-up, you will almost always find yourself donning several hats. While managing everything is relatively easy when you are small, it can become a daunting task to manage things when you start growing rapidly. Multitasking becomes a real skill as you negotiate with clients, send proposals and work on current assignments. With all this chaos, you certainly don't want to miss out on payments – after all, that's what you're working for!

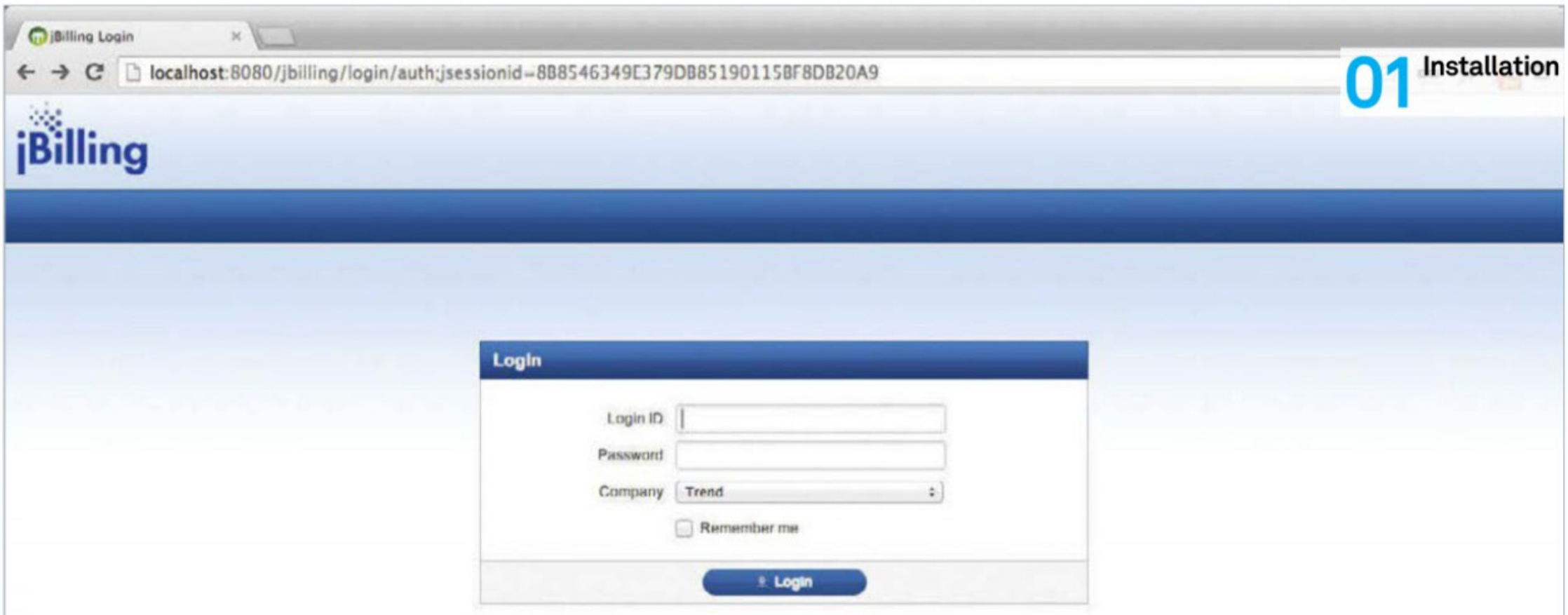
Today we introduce jBilling, which can help you manage the most important aspect of your business – the income. This is not the typical invoice management kind of tool, rather a full-fledged platform with several innovative features. jBilling helps you manage invoices, track payments, bill your customers and more with little effort on your behalf – just what you want when juggling responsibilities.

In this tutorial we will first cover the necessary steps to install and set up jBilling before having a closer look at the various features that can help you manage your business better. We have used the latest stable community edition of jBilling, version 3.1.0, for demo purposes in this article.

Build a billing platform for your business with jBilling

Make managing invoices, payments and billing simple and stress-free

TUTORIAL



01 Installation

jBilling is integrated with the web server out of the box, which helps make the installation process straightforward. Just unzip the downloaded zip file to a folder (where you want the installation to be done), eg 'my_jBilling'. Open the command prompt and navigate to the folder /path/my_jBilling/bin. Assign executable permissions to all the shell script files, with the command **chmod +x *.sh**. Also, remember to set the **JAVA_HOME** variable with your Java Home path. You can then start jBilling by running **./startup.sh**. This completes the installation process – note that the process may slightly differ depending on the OS you use. As the startup.sh script executes, the command prompt shows five lines of logs indicating successful start. You can then access jBilling via your browser at **http://localhost:8080/jbilling** and login with credentials admin/123qwe. You can also access **http://localhost:8080/jbilling/signup** to create your new signup.

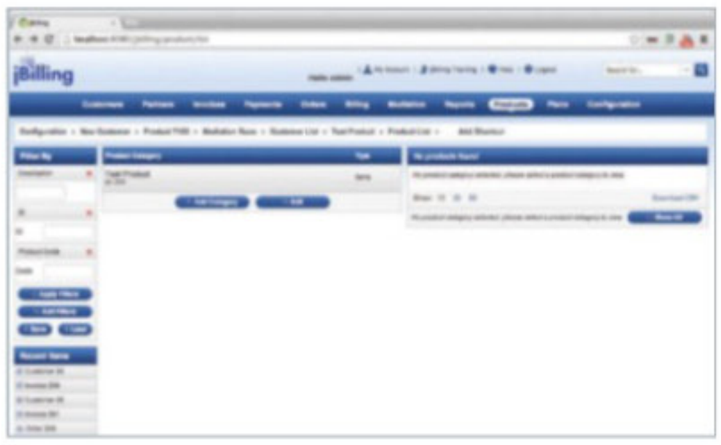
02 Customers

No one wants to add a customer's detail to the system every single time an invoice is sent to them! It is generally a good idea to keep the details of your customer with you and that's precisely what jBilling lets you do – simply click on the 'Customer' button on the main menu to go to the

customer page. Here you can view all the details related to the customer – but before that, you need to add a customer. To do so, click on the 'Add New' button and then fill in all of the relevant details. Note that once you add a customer, a separate login for the customer is also created and they can then log in to your jBilling system and manage their account as well (to make payments, view invoices and so on). This may seem trivial for smaller organisations with a smaller number of customers, but if you have a huge customer base and would like customers to handle payments themselves, you will definitely like this feature.

03 Products

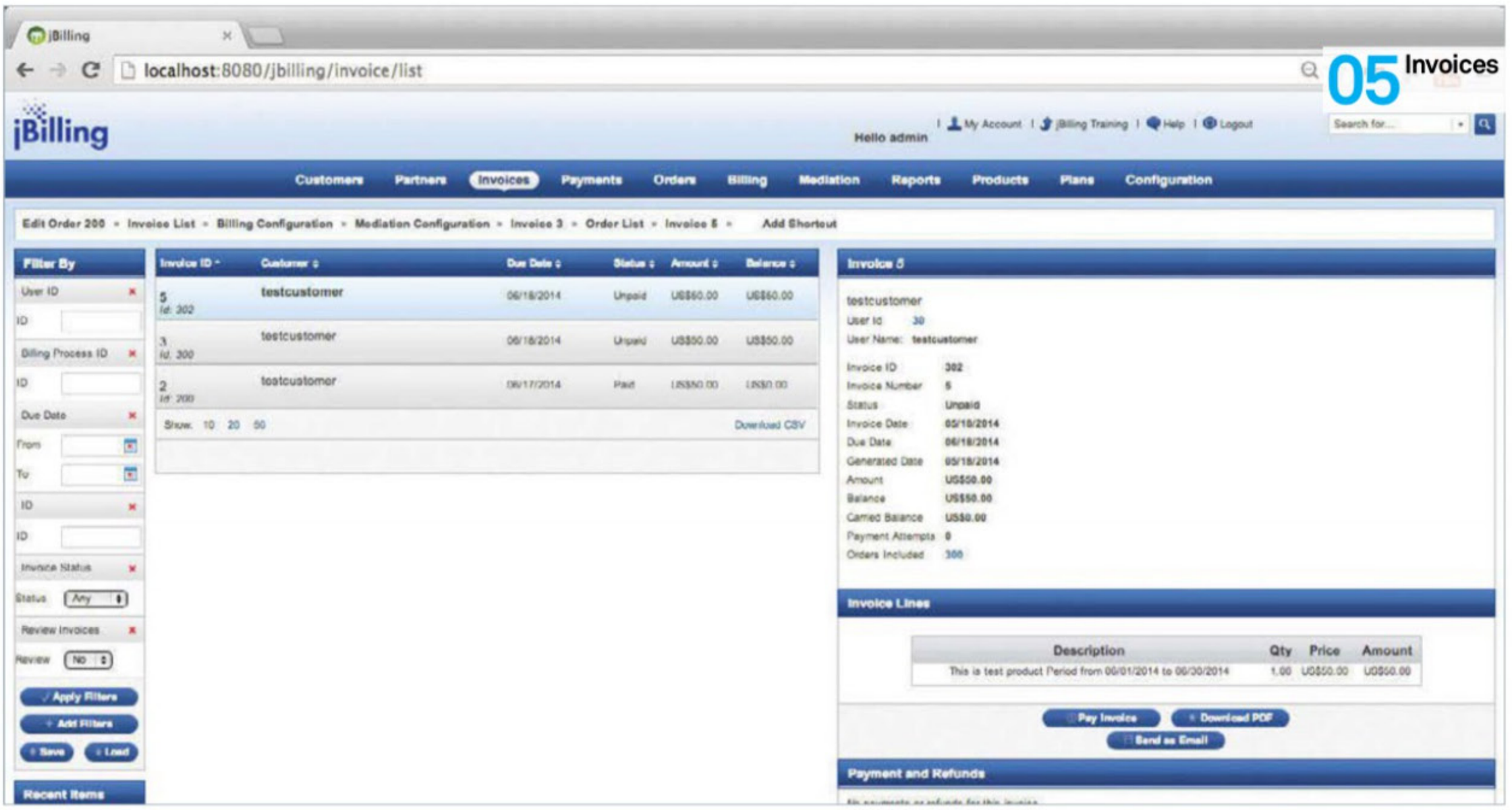
Besides customers, the other important aspect of a business is what you sell – your products or services. Handling your products in jBilling is nice and straightforward. Simply click on the 'Products' button to go to the products page.



To add a new product here, you must add product categories first – click on the 'Add Category' button to do that. After the category is created, select it to add new products to that particular category or view all the products within it. Once you have all your products listed in the system, you can use them to create orders, invoices and so on.

04 Orders

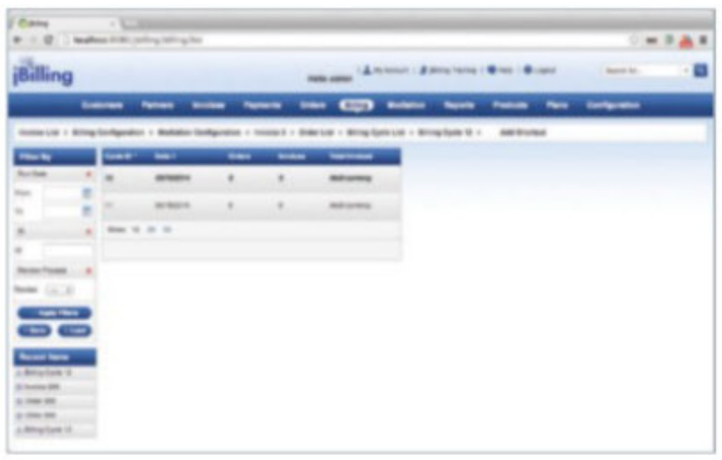
Before serving your customer you need an order from them. jBilling lets you handle orders in a way that closely resembles real-world scenarios. Clicking on the 'Orders' link on the main menu will take you to the orders page where you can view a list of all the orders received up to now. At this point you may be puzzled; unlike other pages there is no button to create an order here.



To create an order you must first navigate to the particular customer you plan to create it for (in the customer page) and then click the 'Create Order' button (located below the customer details). This arrangement makes sure that there is tight coupling between an order and related customer. Once the order is created you can see it in the Order page. You can then edit orders to add products or create invoices out of it.

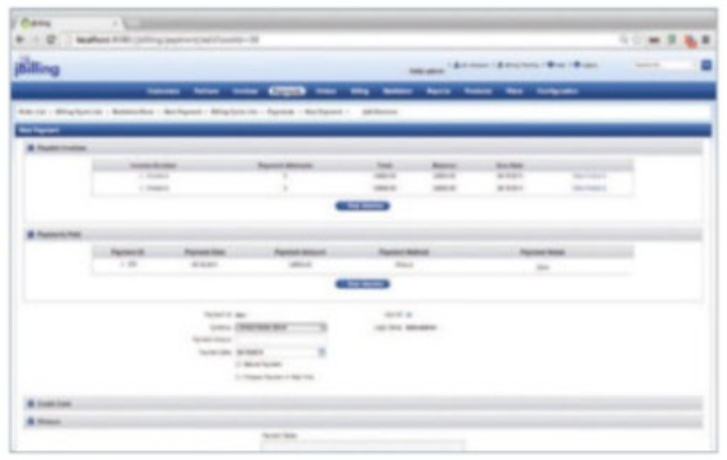
05 Invoices

We have tight coupling with customers and orders, so it makes sense that invoices in jBilling should be related to an order too. So, to create an invoice you need to go to the order for which you are raising the invoice and click the 'Generate Invoice' button. The invoice is then created – note that you can even apply other orders to an invoice (if it hasn't been paid). Also, an order can't be used to generate an invoice if an earlier invoice (related to it) has already been paid. Having generated the invoice, you can send it via email or download it as a PDF. You may find that you want to change the invoice logo – but we'll get to configuration and customisation later on. We will also see in later steps about how the payments related to an invoice can also be tracked.



06 Billing

Billing is the feature that helps you automate the whole process of invoicing and payments. It can come in handy for businesses with a subscription model or other cases where customers are charged in a recurring manner. To set the billing process, you need to go the Configuration page first. Once you are on the page, click on 'Billing Process' on the left-hand menu bar to set the date and other parameters. With the parameters set, billing process runs automatically and shows a representation of the invoices. This output (invoices) needs to be approved by the admin – only once this has happened can the real invoices get generated and delivered to the customer. The customers (whose payments are not automatic) can then pay their bills with their own logins.



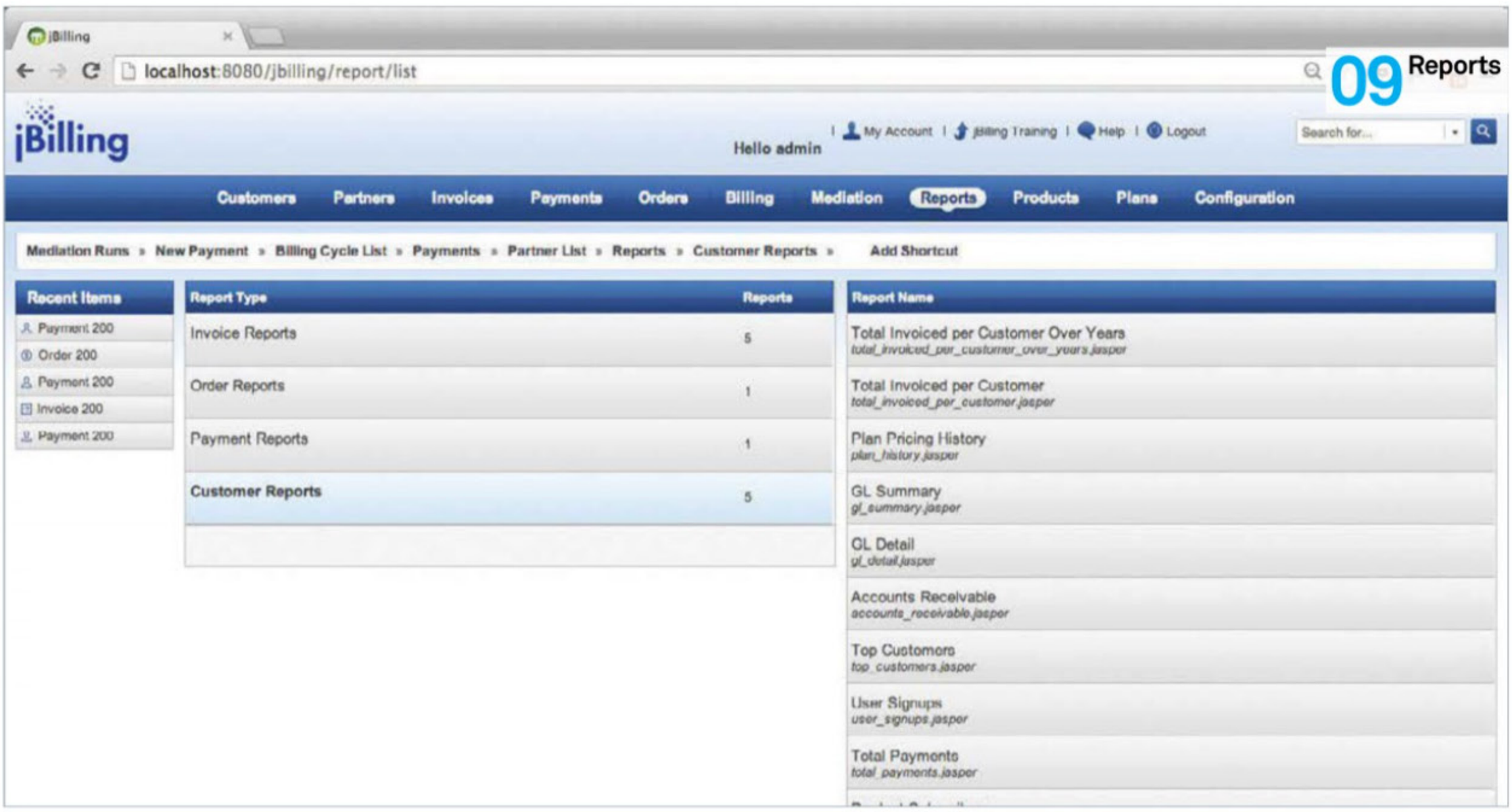
07 Payments

Any payment made for an invoice is tracked on the Payments page, where you can view a list of all the payments already taken care of. To create a new payment, you need to select the customer (for whom payment is being made) on the Customer page and then click the 'Make Payment' button at the very bottom (next to the 'Create Order' button). This takes you to a

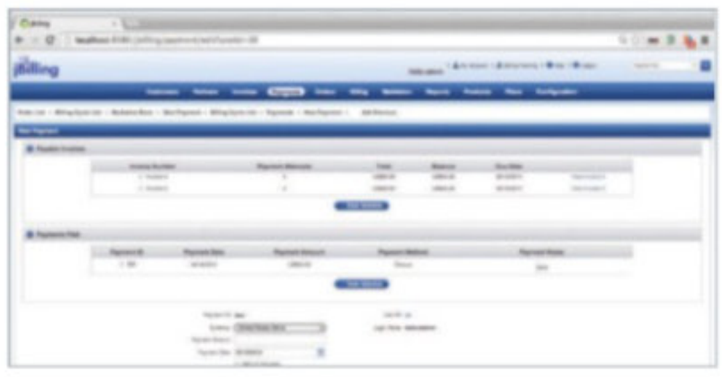
Build a billing platform for your business with jBilling

Make managing invoices, payments and billing simple and stress-free

TUTORIAL



page with details of all the paid/unpaid invoices (raised for that customer). Just select the relevant invoice and fill up the details of payment method to complete the payment process. Later, if there is a need to edit the payment details, you need to unlink the invoice before editing the details.



08 Partners

Partners – for example, any affiliate marketing partners for an eCommerce website – are people or organisations that help your business grow. They are generally paid a mutually agreed percentage of the revenue they bring in. jBilling helps you manage partners in a easy, automated way. Click on the Partners link on the homepage to reach the Partners page and set about adding a new partner. Here you will need to fill in the details related to percentage rate, referral fee, payout date and period and so on. Now whenever a new customer is added (with the Partner ID field filled in) the relevant partner gets entitled to the commission percentage (as set during adding the partner) and the jBilling system keeps a track of the partner's due payment. Note that, as with customers, partners also get their own login once you add their details to jBilling. It is up to you to give them the login access, though.

> **"The reporting engine of jBilling lets you have a bird's-eye view of what's going on with your accounts"**

09 Reports

The reporting engine of jBilling lets you have a bird's-eye view of what's going on with your company's accounts. Click on the Reports link on the main menu; here there are four report types available – invoice, order, payment and customer. You can select one to reveal the different reports available inside that type. After a report is selected, you can see a brief summary of what the report is supposed to show. Set the end date and then click on the 'Run Report' button to run the report. Having done this, the system shows you the output. You can also change the output format to PDF, Excel or HTML.

10 Configuration

The configuration page lets you fine-tune your jBilling installation settings. Click on the Configuration link and you will see a list of settings available on the left menu bar. The links are somewhat self-explanatory but we'll run through the more useful ones. The Billing Process link allows you to set the billing run parameters. You can change the invoice logo using the Invoice Display setting. To add new users, simply click on the 'Users' link. To set the default currency or add a new currency to the system, click on the 'Currencies' link. You can even blacklist customers under the 'Blacklist' link. You will find many more settings to customise jBilling as per your tastes and requirements – just keep exploring and make jBilling work for you.

Create a two-step authentication with Twilio

Increase security in access to your web services by building a simple two-step authentication with Twilio's SMS APIs to help you

Advisor

Sean M Tracey is a creative technologist at a leading digital agency. He spends a lot of his time living inside of Node.js, Python and Arduino

Resources

Python 2.7+

Flask 0.10.0:
flask.pocoo.org/

Flask Github:
github.com/mitsuhiko/flask

A Twilio account:
twilio.com

Twilio's Python REST API Helper Library:
github.com/twilio/twilio-python/zipball/master

MySQLDB:
mysql-python.sourceforge.net

Telephony is one of the most versatile technologies in our households. Despite being invented over 100 years ago, we still use the same basic infrastructure that once only carried the voices of people to deliver a rich multitude of media content at incredible speeds. As is often the case with wonderful things, they can often be complex too – and yet phones are more important now to our daily lives than ever. So, what can we do to leverage some of that versatile technology?

Well, for starters we can use an API. Twilio has created a RESTful API that removes a great deal of that complexity of telephony so that we can write apps and services that are able to deliver and receive both phone calls and SMS using various endpoints and services. Neat! In this tutorial, we're going to look at using Twilio to help us create the basic flow for a two-step authentication system for logging into a service. We're also going to be using Flask to help us create our routes and generate our pages, but little of Flask's detail will be covered here.

The Twilio interface is kept nice and simple – no unnecessary complications here

01 Get a Twilio account and phone number

Signing up to Twilio is pretty easy. First, head over to **http://twilio.com** and click the 'Signup' button. At this point, the sign-up process doesn't really differ from any other service, but after you've entered an email address and password you'll be asked for a phone number. Given the nature of Twilio's API, it makes sense for them to ask whether we're human, and having them text us is a good way to confirm that. Hey, it's a two-step authentication, which is exactly what we're working towards.

You can enter any number you have access to, be it a landline or mobile, to confirm who you are, but at this point we suggest you authenticate using a phone that can accept SMS (instead of a landline). Having entered your number, you'll receive a text to authenticate your phone – enter it and you'll be presented with a Twilio phone number. This is your Twilio phone number and you'll be using it to send and receive our authentication texts.

02 Add credit

Just like a mobile phone operator, Twilio is not a free service – although it is very inexpensive. In order to continue, we'll need to add a card and some funds to our newly created Twilio account. On the main page of the dashboard, you'll see a big blue dialog asking to upgrade your trial account; click through and follow the instructions to add a card and the amount of credit you would like to use. The minimum amount of $20 (around £10 GBP) will be more than plenty for this and other projects. Once that's done, you're almost ready to start sending text messages – but first head back over to the Twilio dashboard and copy your account SID and auth token down somewhere, you'll need those a little later.

Create a two-step authentication with Twilio

Increase security in access to your web services with Twilio's SMS APIs

TUTORIAL

03 Install the Twilio Helper Library and MySQLDB

The Twilio helper library is a fantastic piece of code that lets you jump straight into sending and handling text messages in no time at all. There are a couple of ways to install the library: you can use either PIP or Easy_Install, like so

```
$ pip install twilio
$ easy_install twilio
```

Or you can download the source code for the helper library and run the 'setup.py' file. It really is as simple as that. Now, for storing the verification tokens we're going to use a MySQL database. To get Python talking to our SQL server, we'll use the Python module MySQLDB, the package for which you can grab like so...

```
apt-get install python-mysqldb
```

In the tutorial resources we have included an SQL dump with the table structure. Import it into a database of your choosing. Assuming everything so far has gone swimmingly, you can create a new project folder/environment and add a new file 'server.py'.

04 Server setup

Open the 'server.py' file for editing. The first thing we're going to do is import the libraries we need for our authentication flow, create the endpoints for our server and assign some of the variables needed to run our Flask server. **(Fig 01)**

You may have noticed the **account_sid** and **auth_token** variable we've set after the import statements. We'll use these with our Twilio client so we can interact with Twilio and our mobile phones. These settings can be found on the Twilio account dashboard, right below the header. We've also connected to our SQL database, so make sure your SQL server is running before you fire up the app, otherwise you'll have an error thrown. Save, now if you run your 'server.py' file, you should be able to access the index page of your server at 127.0.0.1:5000/.

05 Server logic

If you've hit all of your server endpoints already, so far all you will see are the strings we returned at the end of of endpoint declarations. These are not all that good-looking, so let's add some Flask templates to pretty things up a little. The focus of this tutorial is not on the intricacies of Flask and as such, included on the DVD is a folder called 'templates' and another called 'static'; copy them both to the root of your current project folder and amend your endpoints as in **Fig 02** (page 47).

If you revisit the pages again, things might seem a little out of whack at the moment, but don't worry about that for the time being. It's mostly because we've not put together the server logic to help the templates figure out what to do. Let's deal with the '/' path first. All we're doing here is checking the state of our session cookies and effecting how the index.html page renders according to that state. If the user isn't logged in, we'll give them a link to the login page, if the user is logged in but hasn't verified, then we'll give them a link to the code verification page. Before we deliver the template we need to check that our session has its particular variables set, otherwise we'll end up getting KeyErrors.

Fig 01

```
import MySQLdb
from flask import Flask, redirect, request, session, render_template
from twilio.rest import TwilioRestClient as twilio
import string, random, time

db = MySQLdb.connect(host="127.0.0.1", user="SQLUSER", passwd="SQLPASS", ↵
db="two-step", port=3306)

expirationLength = 300

account_sid = "YOUR ACCOUNT SID"
auth_token = "YOUR ACCOUNT AUTH TOKEN"
client = twilio(account_sid, auth_token)

@app.route('/')
def index():
    return "index page"

@app.route('/login', methods=['GET'])
def login():
    return "login page"

@app.route('/check-user', methods=['POST'])
def checkUser():
    return "check user page"

@app.route('/logout')
def logout():
    return "logout page"

@app.route('/verify', methods=['GET'])
def twoStep():
    return "verify page"

@app.route('/check-code', methods=['POST'])
def checkCode():
    return "check code page"

if __name__ == '__main__':

    app.secret_key = 'R4nDOMCRypt0gr4ph1cK3yf0R5355i0N'
    app.run(host='0.0.0.0', debug=True)
```

```
@app.route('/')
def index():

    checkSessionState()

    return render_template('index.html')

def checkSessionState():
```

»

Create a two-step authentication with Twilio


```
try:
    session['verified'] == True
except KeyError:
    session['verified'] = ''

try:
    session['loggedin'] == True
except KeyError:
    session['loggedin'] = ''

try:
    session['user'] == True
except KeyError:
    session['user'] = ''
```

06 Logging in

The first step in two-step authentication is logging in with a traditional username/email and password. Access your database and create a new user with the following query:

```
INSERT INTO users (username, password, phonenumber) VALUES ('A USERNAME', 'A PASSWORD', '+44YOURUSERSPHONENUMBER')
```

For the purposes of this tutorial, the password is plain text – but we implore you, when you're implementing passwords in a live environment, make sure that you hash them. Still, for now we're going to use a plain text password. Our login.html template has a form that's going to POST itself to check-user; here we'll check the validity of the credentials and then trigger the verification if needed. So we're going to use the MySQLDB module to get details from our database.

In order to query our database we need to create a cursor from which to execute our MySQL statements. We do this with **cur = db.cursor()**:

```
@app.route('/check-user', methods=['POST'])
def checkUser():

  #session.clear()

  if request.method == 'POST':
    #print request.form['username']

    cur = db.cursor()

    cur.execute("""SELECT * FROM users WHERE username = %s""", (request.form['username'],))
    result = cur.fetchone()
    returnedPassword = result[2]
    returnedPhoneNumber = result[3]
```

We can then build an SQL statement using **cur.execute()**. Notice the **%s**; this will be replaced with the value passed through in the next variable. We execute the statement with **cur.fetchone()**, which will get us the first row that the query returns – if there is no result we'll get None and we can then return the user to the login page with an error message. Let's assume we've requested a valid user – we'll next check that the password assigned to that user is the same as the one submitted. If so, we'll generate the validation code to send to the user, which we'll store in the verification table of our database until it's used or expires. We'll need to create a new cursor to insert the verification code into our table. After we've executed the statement we need to commit the changes to the database, we do this with **db.commit()** – we'll then add the results of the query to our session so we can check against them later. **(Fig 03)**

07 Send the verification code

Now that we believe we're dealing with a valid user, it's time for the second step of our two-step process. On the line after where we stored a variable in our session, we make a call to **sendVerificationCode (VERIFICATION CODE, USER PHONE NUMBER)** and pass through the code we want to send to our user and the user's phone number. So what does that function actually look like? It must be big, long and complicated because it deals with the telecommunications network, right? Wrong. It's actually incredibly simple to send an SMS with Twilio. In fact, part of the inherent beauty of Twilio lies in its simplicity. To send a text, all we have to do is:

```
def sendVerificationCode(code, number):

  text = client.messages.create(
    body="Your verification code is:" + code,
      to=number,
      from_="+YOURTWILIONUMBER"
    )

return text.sid
```

Using the **client** variable we used to instantiate the Twilio REST module, we can access the **messages** class and execute the **create** method. All we need to pass through is the text that will make up the body of our message, the number that we want to send it to and the number that we want to send it from. When inputting the number that we want to send it to, it's best to use the +CountryCode type of phone number to avoid any ambiguity about the intended recipient. The number that we're sending from is our Twilio number; you can use any Twilio number you have assigned to your account, so long as it has credit. As soon as we execute that code, the message will be sent and your SMS will go straight through to the telephone. The SID is the unique identifier for the message/call sent; receiving it means the message has been executed successfully. After that, we can redirect our user to the verification page with **return redirect('/verify')** at the end of /check-user.

08 Check verification code

At this point the user will have received a text message with something along the lines of 'Your verification code is: 12cd56' and will be presented with the verification page. If, at this point, they choose to browse around our site, they won't be able to access anything that we don't want them to. Still, we'll know that they've logged in, so if they head back to the verification page, we can just let them input their code. Once they submit their code, it will be sent to the /check-code endpoint.

Just like before when we checked for our user's validity, we're going to attempt to retrieve the verification code and check it. **(Fig 04)**

First we're simply going to retrieve the code and check the user it has assigned to it. If that user assigned to the code matches the user in our session, then we can be certain that the right person is logging in with the right code – if not we can redirect them accordingly. Assuming the code is correct, we need to check it's still valid. Back in Step 6, we created an expiration time that was five minutes in the future from when the code was generated. If it's been more than five minutes (or whatever time you've set on it) then we're going to consider it invalid, delete the code from our table and then log out our user so they can start over, like so.

```
elif time.time() > expirationTime:

  expirySQL = db.cursor()

  expirySQL.execute("""DELETE FROM verification WHERE code=%s""", (codeToCheck,))

  expirySQL.close()
```

Create a two-step authentication with Twilio

Increase security in access to your web services with Twilio's SMS APIs

TUTORIAL

```
session['loggedin'] == False

return redirect('/logout')
```

If we manage to pass the tests so far, then we've two-step verified our user – hooray! Surprisingly easy, eh? Before we give our user free reign around our service, we still want to get rid of that token – we don't need it any more and we don't want to risk someone else using it maliciously in the future.

```
else:
  delSql = db.cursor()

  delSql.execute("""DELETE FROM verification WHERE code=%s""", (codeToCheck,))

  delSql.close()

  db.commit()

  session['verified'] = True

  return redirect('/')

else:
  return redirect('/verify?error=true')
```

And that's it! Now we redirect our user to wherever we want them to be at the end of the process. In this instance we're sending them back to our index page, which will render a success message and give the user a link to log out whenever they like – but they could be redirected to their user page, and so on.

09 Conclusion

In every web-based service, security is king. Users entrust more and more personal data and trends to services every day and it's the responsibility of those services to maintain the privacy of that data as best they can. It's no wonder that services such as Amazon, Google and Facebook have all implemented two-step verification across their services. With two-step authentication, a user can tie their account to one of the most personal things they own: their phone. In recent times, mobile technologies have come to contain people's lives; now with services like Twilio and some simple code, they contain people's keys, too – or at least a part of them.

Fig 02

```
@app.route('/')
def index():
  return render_template('index.html')

@app.route('/login', methods=['GET'])
def login():
  return render_template('login.html')

@app.route('/check-user', methods=['POST'])
def checkUser():
  return "check user page"

@app.route('/logout')
def logout():
  return "logout page"

@app.route('/verify', methods=['GET'])
def twoStep():
  return render_template('verify.html')

@app.route('/check-code', methods=['POST'])
def checkCode():
  return "check code page"
```

Fig 03

```
verficationCode = generateVerificationCode(size=6)

ins = db.cursor()

expiration = int(time.time() + expirationLength)

sql = "INSERT INTO verification (code, expiration, username) VALUES ('%s', '%s', '%s')" % (verficationCode, expiration, request.form['username'])

ins.execute(sql)

ins.close()

db.commit()

session['user'] = request.form['username']
session['loggedin'] = True
```

Fig 04

```
@app.route('/check-code', methods=['POST'])
def checkCode():

  if request.method == 'POST':
    codeToCheck = request.form['code']

    if not 'user' in session:
      return redirect('/login')
    else:
      cur = db.cursor()

      cur.execute("""SELECT * FROM verification WHERE code = %s""", (codeToCheck,))

      result = cur.fetchone()

      cur.close()

      if result != None:
        returnedUser = result[3]
        expirationTime = int(result[2])

        if returnedUser != session['user']:
          return redirect('/verify?error=true')
```

Code Android apps with Java

There are untold riches awaiting those who can create smartphone apps – luckily getting started with Android is really easy

Advisor

Tam Hanna has followed the ups and downs of the mobile market since the days of the now-legendary Palm IIIc. His firm has managed news sites, developed applications and acted as consultant.

Developers seeking to create apps for Google's popular smartphone operating system needn't bother themselves with learning complex and often obscure custom versions of C++ like the one used by the once-famous Symbian OS. Instead, Android applications tend to be written in Java, with native modules added in for convenience and speed.

This ease of development has led to a flood of apps hitting the Play Store, so making a living from a well-made app is not as easy as it used to be. The large selection means that well-planned marketing and a great-looking user interface are now of paramount importance.

Creating apps, however, is not limited to people seeking to publish their products in app stores. Instead, a custom app can also be helpful when it comes to solving everyday tasks. For example, let's say there's a person working as a trader. Their stockbroker recently introduced a new feature that could save them a lot of time, but it requires some complex mathematical computations. Creating an app that solves the equations on the fly would be pretty useful here.

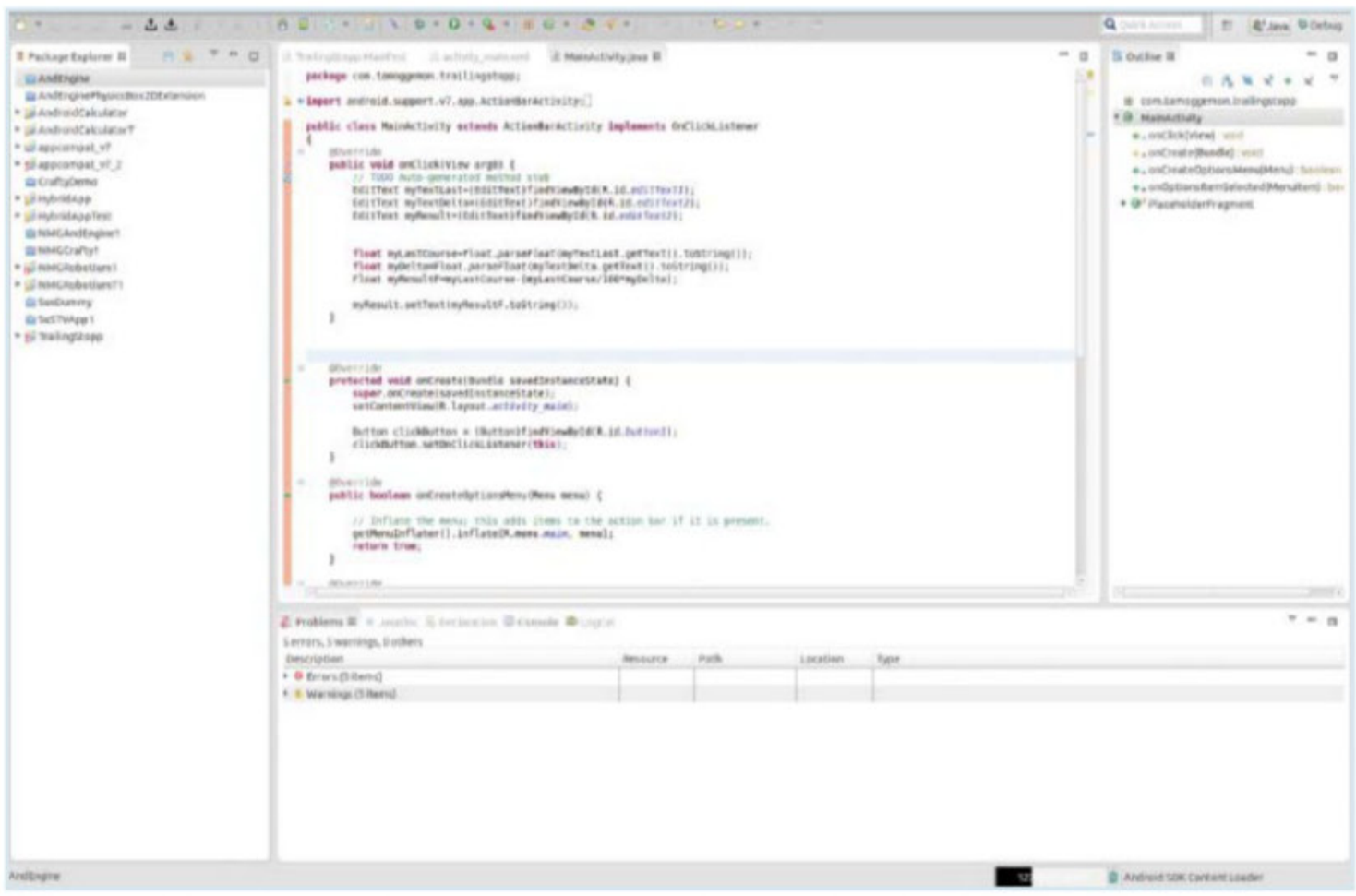
Even though such a program is unlikely to sell on Google Play, having it on hand can save a lot of time. From a workflow point of view, creating this application is not that different from the process used when building a larger product.

Further reading...

www.vogella.com/tutorials

https://developer.android.com/

http://android-developers.blogspot.sk



Java mode is used for editing the source code of your application

Find me tools!

Developing applications for mobile phones is done with a cross-compiler. Fortunately, operating system vendors provide software development kits that contain all of the necessary modules developers require. In the case of Android, the product is based on the well-liked Eclipse code editing system.

Usually, Eclipse users need to assemble their toolchain out of individual building blocks. Google solves this problem by providing users with a ready-made package containing the integrated development environment, the necessary plug-ins and an emulator.

Most of the tools are based on Java. This means that your machine needs to be equipped with a Java development kit – installing it is as straightforward as entering the following command into a terminal window:

```
sudo apt-get install openjdk-6-jdk
```

When working with a 64-bit version of Linux, additional packets need to be installed. More information on the specific instructions needed by the various distributions can be found through a quick search if need be.

Code Android apps with Java

Build bespoke smartphone tools to solve specific tasks

TUTORIAL

Version mayhem

Stay on top of changing OS

Google changes its operating system frequently. Each release is identified by its API level, which is described by the text API followed by a number. In order to maximise their possible customer base, developers should not limit their applications to the very latest version of the operating system.

The value set in 'Minimum Required SDK' identifies the minimal version of Android that is required for starting the application. Your program will not work on older handsets.

Target SDK is an 'internal' information for the operating system. It specifies which version of the operating system was used for testing the application – later versions will use this setting in order to configure their compatibility libraries.

Finally, the 'Compile with' setting decides which version of the toolchain is to be used for compiling. Android is mostly downward compatible, so developers should usually select the latest version of the SDK available.

Fig 01

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.tamoggemon.trailingstopp"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="19" />
```

Fig 02

```
<application
  android:allowBackup="true"
  android:icon="@drawable/ic_launcher"
  android:label="@string/app_name"
  android:theme="@style/AppTheme" >
  <activity
    android:name="com.tamoggemon.trailingstopp.MainActivity"
    android:label="@string/app_name" >
    <intent-filter>
      <action android:name="android.intent.action.MAIN" />

      <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
```

In the next step, visit Google's SDK download page, located at **http://developer.android.com/sdk/index.html**. Scroll down until you find a table labelled ADT Bundle and click the corresponding file in order to download it to your machine. Extract it to a convenient location of your choice – keep in mind that development environments are IO-heavy, so it is recommended that you use the fastest disk installed on your machine.

Your zip file is made up of two subfolders. /eclipse/ contains the files needed for the graphical development environment, whereas /sdk/ is made up of system files that usually don't need to be modified. Starting the Android Developer Tools can be accomplished by running the 'eclipse' file found in the /eclipse/ subfolder. Eclipse will ask you for a working directory. This folder will contain both settings and projects – back it up in order to save your entire workspace.

A first project

Starting Eclipse can take a minute or two. Once the process is completed, the product will present you with a screen similar to the one shown across the page. The two toggles at the top right-hand side of the display allow you to change the mode: click 'Debug' in order to activate the screen for finding errors. Then, click 'Java' to return to editing.

New projects can be created by opening the generation wizard found under New>Android Application Project. In the first step, the IDE will ask you for further information about the name, package and versioning of the project in question (see boxout).

Our example will be called TrailingStopp, and is located in the package 'com.tamoggemon.trailingstopp'. Google provides reasonable values for the rest of the wizard, which can be completed by clicking Next multiple times. When done, a new project will have been created.

Android applications can be broken down into three types of file. First of all, resources are stored in XML format. 'AndroidManifest.xml' has special priority in this regard as it acts as an application descriptor, providing the operating system with further information about the app. You can see our example structure in **Fig. 01**.

Manifest files start out by declaring the SDK version and the package names required by the application. This information enables the operating system to assess whether the application is compatible with the current hardware.

In the next step (**Fig. 02**), the actual application is declared along with any activities. Google opted to call forms Activities and we will get to those in a minute.

Interprocess communication takes place via Intents. Our application informs the system that the LAUNCHER-Intent is to be handled by the Activity MainActivity. Finally, the open tags are closed in order to create a valid XML file.

```
    </activity>
  </application>

</manifest>
```

Even though the manifest can be edited with a graphical editor, developers should be aware of its role and structure. Many advanced features can only be activated by hand. Furthermore, most tutorial authors simply provide their readers with a summary of the necessary changes to the file structure.

Activities and Layouts

By default, a newly-created application consists of but one form. In our case, the activity's structure is declared in a file called ActivityMain.xml. Double-clicking it in the Package explorer usually opens a WYSIWYG editor such as the one shown over the page. Should your version of Eclipse display code, just click the little tab labelled 'Graphical Layout' to fix the problem.

Fig 03

```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:id="@+id/container"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context="com.tamoggemon.trailingstopp.MainActivity"
  tools:ignore="MergeRootFrame" >

  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
```

> Every Android phone can be used as a debug target

Fig 04

```
    <EditText
      android:id="@+id/editText1"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:ems="10"
      android:inputType="numberDecimal" >

      <requestFocus />
    </EditText>

    <EditText
      android:id="@+id/editText2"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:ems="10"
      android:inputType="numberDecimal" />

    <EditText
      android:id="@+id/editText3"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:ems="10"
      android:inputType="numberDecimal" />

    <Button
      android:id="@+id/button1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Button" />

  </LinearLayout>

</FrameLayout>
```

The Palette window on the left side of the form contains controls that can be added by dragging and dropping. Eclipse's rendering is relatively accurate – things like layouts get considered during the generation of the preview.

As outlined in the callout, our algorithm requires a total of three input parameters. Collecting data tends to be done with text fields and Android's GUI stack contains a good dozen text fields. We will use a control that is limited to the input of decimal numbers – the Palette lists it as '42.0'; its tooltip calls it Number (Decimal). In an ideal world, we could simply drag the controls onto the form and be done with it – but Android phones are offered with a large variety of screen sizes. Google addresses this problem via a layout system that was inspired by the one used in Qt.

Arranging our widgets above one another requires us to use a **LinearLayout** (vertical). Thus, your first change involves dragging said control into the form area. Due to their invisible nature, the layout will not show up in the preview. Should you ever need to find it, use the Outline window in order to track it down.

In the next step, the three text boxes can be dragged in. Eclipse will automatically arrange them above one another. Clicking any of them allows you to edit its properties in the list on the bottom right-hand side of the screen. We would recommend that you change the names of the widgets and provide them with a Hint text that will show up if the user has not entered data. Very diligent individuals can go further, adding in three labels describing the data to be entered in the text box below. Finally, drag in a button. It will be used for triggering the actual computation process. The resulting XML file will look like **Fig. 03**.

Controls are assigned to layouts by enclosing them between the opening and closing tag. Feel free to nest layouts as deep as you see fit – but please keep in mind that the amount of layouts directly affects the performance of the GUI stack.

The actual widgets are declared in a HTMLesque fashion. The string assigned to **@id** is used for finding the controls in the source code: by using a string that starts with a plus, we 'add' the name of the widget to the list of elements stored in **id**. (**Fig. 04**)

Change the code

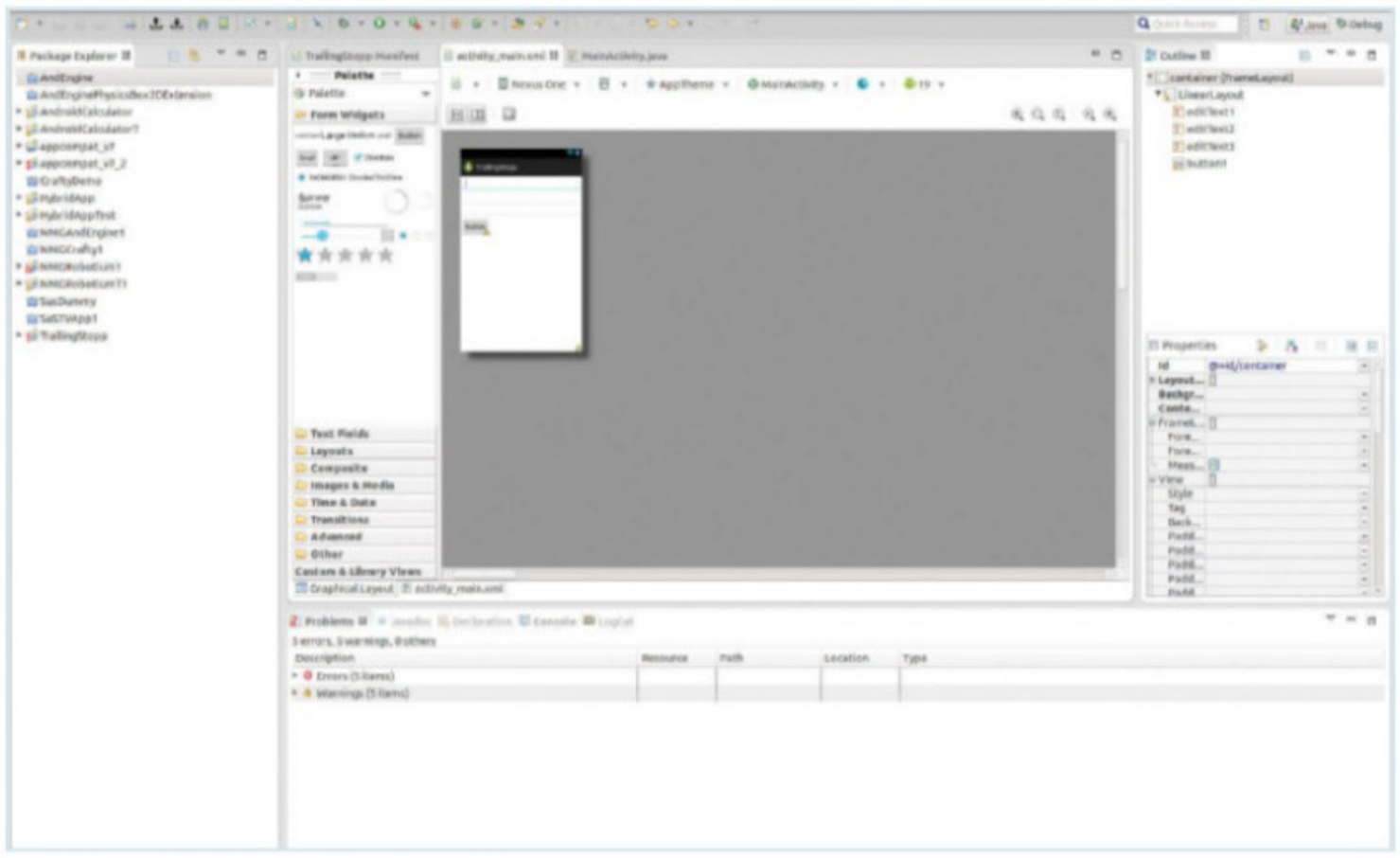
With that, the resource file is ready. Its intelligence can be found in the Activity class, which resides in '/src/<packet>/MainActivity.java'. Our application wants to be notified

whenever the button is pressed – this can be accomplished by changing the body of the **onCreate()** method. (**Fig. 05**)

Android instantiates activities by providing them with XML content – this is handled in the **setContentView** function that had already been provided by the framework. The next step involves finding the button by using the ID that was assigned to it in the XML file we mentioned above.

onClickListener needs to implement an interface. Eclipse will offer autocompletion tools – using them leads to the skeleton structure found in **Fig. 06**. With that, we can proceed to implementing the actual calculation method. The final version of **onClick** looks as in **Fig. 07**.

This routine starts out by retrieving pointers to the three text boxes from the R array. The rest of the code is made up of normal Java code, which is run as is. With that, we are done – our calculator is ready for deployment!



Adding widgets to a form is really easy

Test your app

Sadly, an app that works on paper does not help our friend with the stockbroker. They want to use the app on their phone. Furthermore, real-life programs tend to be complex enough to ensure that some errors will pop up – finding them is much easier if you can see the app working on a phone.

Every Android phone can be used as a debug target as long as it is connected to your PC and has its developer mode enabled. Sadly, Google changed the way the activation works in recent versions of the operating system, so we cannot give you a universally valid sequence. Connect your handset to your PC and click the Play or the Debug buttons – Eclipse will do the rest.

Sometimes, carrying around a phone and its accompanying data cable isn't feasible. If this is the case, Google offers an emulator that provides a pretty decent (albeit very slow) representation of a real handset. It's worth remembering that virtual machines need to be set up before use. This can be accomplished easily by opening up the AVD manager, located at Window>Android Virtual Device Manager.

Conclusion

We covered a lot of important information in this first instalment of our three-part series on Android. However, important topics such as permission management and the activity lifecycle have not yet been discussed. We will cover them in the next issue – may all your apps be the next *Flappy Bird*...

Fig 05

```
@Override
protected void onCreate(Bundle savedInstanceState) {
  super.onCreate(savedInstanceState);
  setContentView(R.layout.activity_main);

  Button clickButton = (Button)findViewById(R.id.button1);
  clickButton.setOnClickListener(this);
}
```

Fig 06

```
public class MainActivity extends ActionBarActivity implements
OnClickListener
{
  @Override
  public void onClick(View arg0) {
    // TODO Auto-generated method stub

  }
```

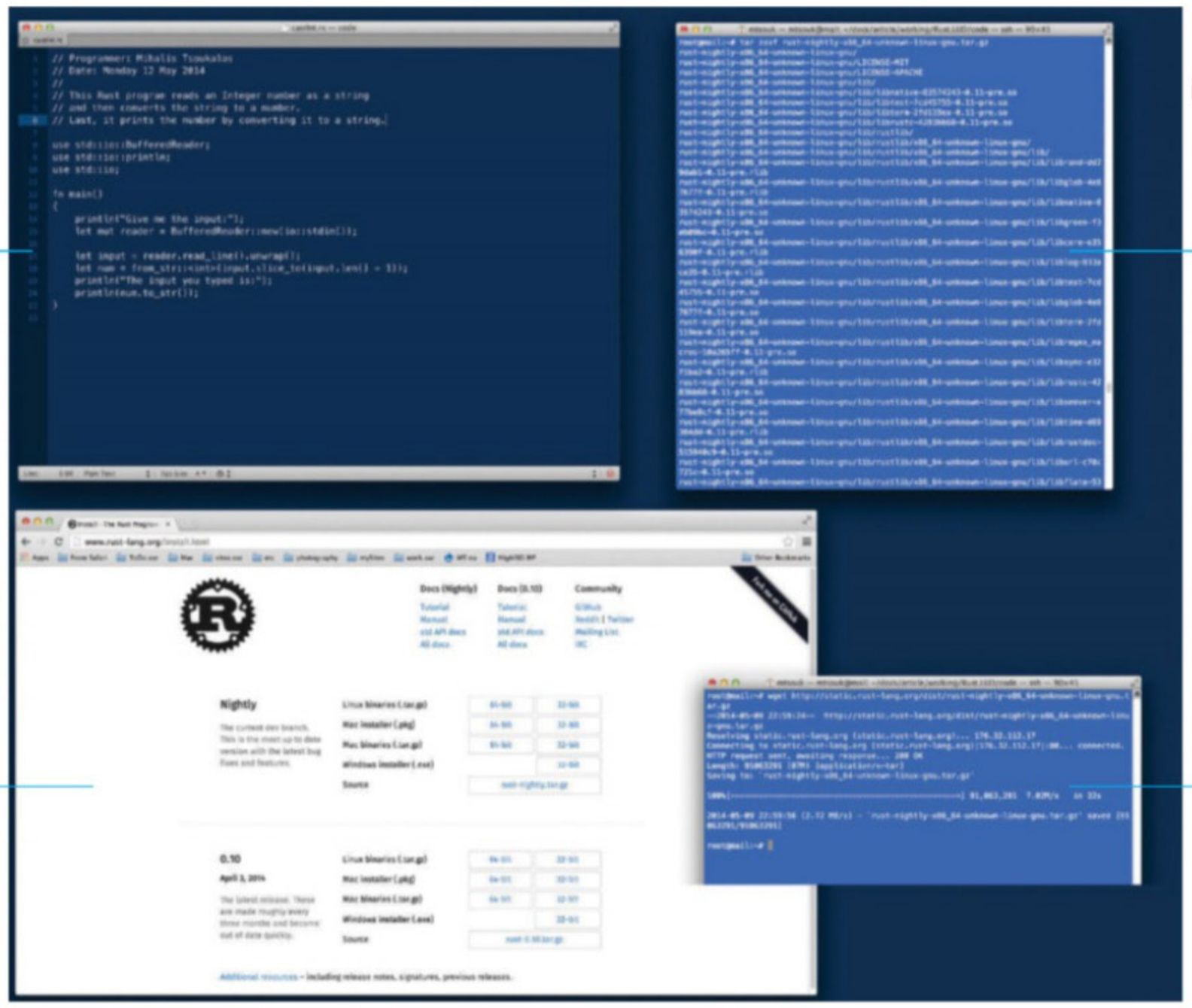
Fig 07

```
public void onClick(View arg0) {
  // TODO Auto-generated method stub
  EditText myTextLast=(EditText)findViewById(R.id.editText1);
  EditText myTextDelta=(EditText)findViewById(R.id.editText2);
  EditText myResult=(EditText)findViewById(R.id.editText3);


  float myLastCourse=Float.parseFloat(myTextLast.getText().toString());
  float myDelta=Float.parseFloat(myTextDelta.getText().toString());
  Float myResultF=myLastCourse-(myLastCourse/100*myDelta);

  myResult.setText(myResultF.toString());
}
```

A small Rust program that converts a string into an integer



We are extracting Rust files from the installation archive before installing

Here you are able to choose your Rust installation package

Here we are downloading the Rust nightly Linux installer

Programming in Rust

Advisor

Mihalis Tsoukalos is a UNIX system administrator also proficient in programming, databases and mathematics. He has been using Linux since 1993

Resources

Rust:
www.rust-lang.org

Rustdoc:
http://static.rust-lang.org/doc/master/rustdoc.html

Getting Rust:
www.rust-lang.org/install.html

Discover Rust, the systems programming language developed by Mozilla that's fast, and wants to be better than C and C++!

Rust is a relatively new systems programming language that aims to avoid unpleasant bugs caused by unsafe code. Those of you who have experience with C or C++ will likely find the syntax of Rust quite familiar.

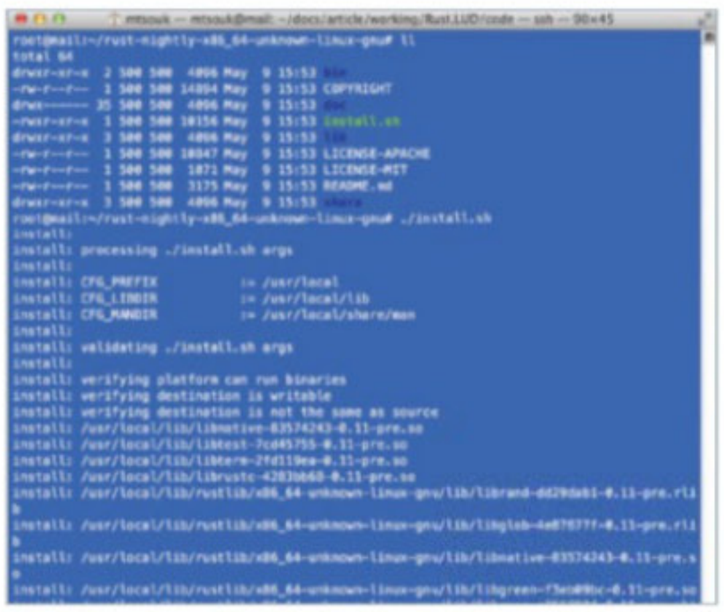
While Rust uses many familiar concepts from other mainstream programming languages, it also borrows elements from research languages including 'owning pointers' and 'borrowed pointers'. Currently Rust is changing quite fast but still, we think the best way to learn programming with Rust is to just get hands-on and try writing some programs. All you need to get going is the Rust compiler and your favourite text editor. The Rust compiler is ingenious and able to find many errors as it's responsible for enforcing the safety rules, preventing your Rust code from compiling until you make the necessary corrections to your code.

If after finishing this article you are hungry for more information, you will find Rust's website (**www.rust-lang.org**) a good place to continue your learning. In the meantime, let's get started with the installation!

Programming in Rust

The systems programming language that's fast and wants to be better than C++

TUTORIAL



01 Install Rust

As Debian 7 does not have a package ready for installation, a manual installation of Rust is required. The Rust site has precompiled binaries (**www.rust-lang.org/install.html**) for most operating systems that you can download. After having extracted the Rust archive, you can then install Rust by simply running the **install.sh** script using root privileges.

The default installation place for the Rust compiler is the /usr/local/bin directory. The name of the compiler executable is rustc. Rust also installs another tool, called **rustdoc**, which is used for generating documentation from all of the Rust source files.

02 Compile your first Rust program

The Rust code for the Hello World program is the following:

```
fn main() {
println!("Hello World!");
}
```

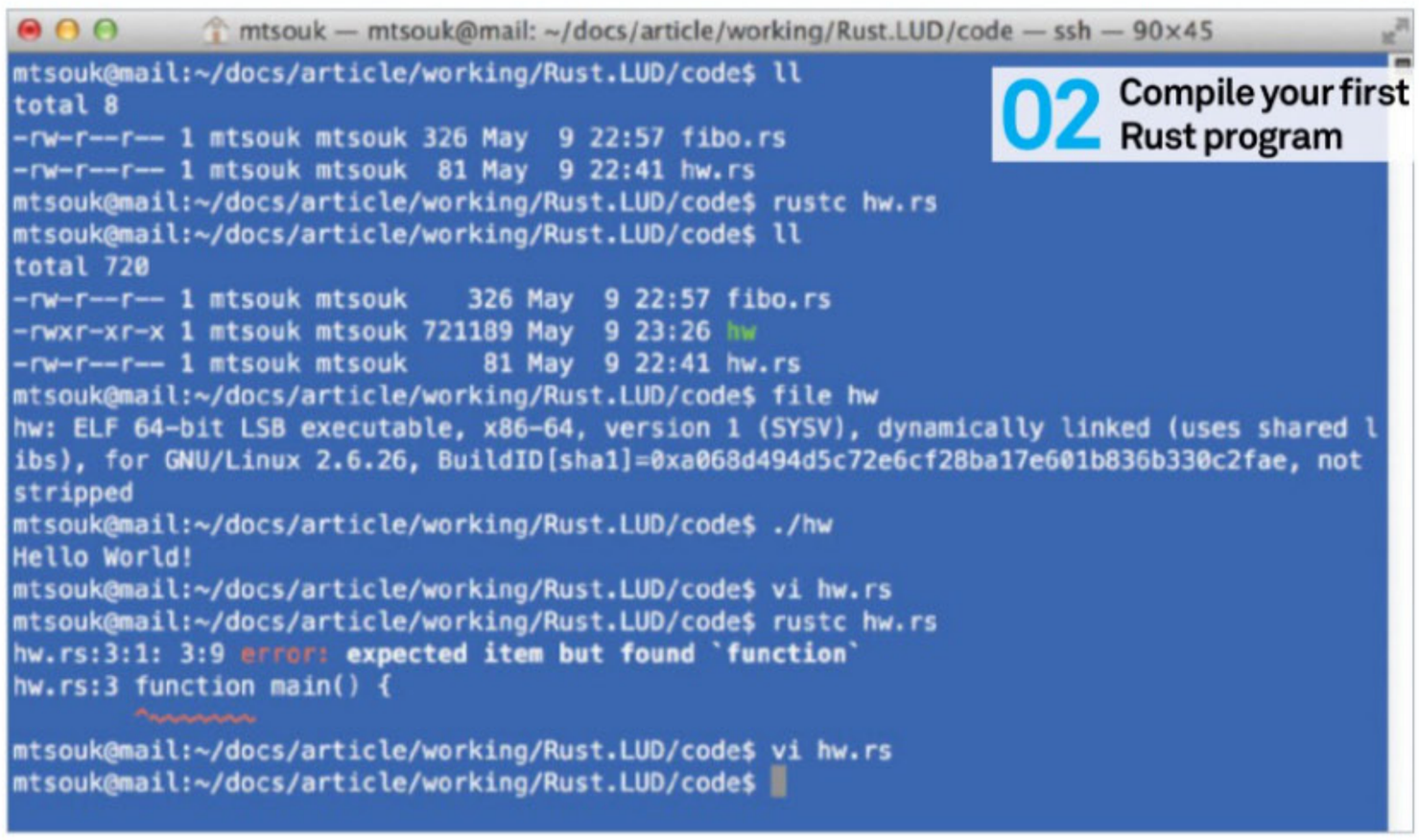
By convention, Rust code is saved in files that have an '.rs' extension, so the "Hello World!" program can be saved as 'hw.rs'. You can compile the hw.rs source file as follows:

```
$ rustc hw.rs
```

If you get the following error messages while trying to compile hw.rs...

```
error: could not exec the linker `cc`: no
       such file or directory
error: aborting due to previous error
```

... it means that you do not have a linker installed. You can solve the problem by installing the gcc package and its dependencies. If everything is okay, Rust will automatically create an executable file called 'hw' that you can run as usual.



02 Compile your first Rust program

> **Rust uses familiar concepts from other programming languages**

03 Basic language features

You can declare a new variable with the keyword **let** and optionally give a type to it. The compiler will then try and find out the type but if it fails, it will give you an error message. It is considered good practice to always give a type to a variable, as they are immutable by default. In order to introduce a local variable whose value you are able to change later, you should use the **let mut** keywords.

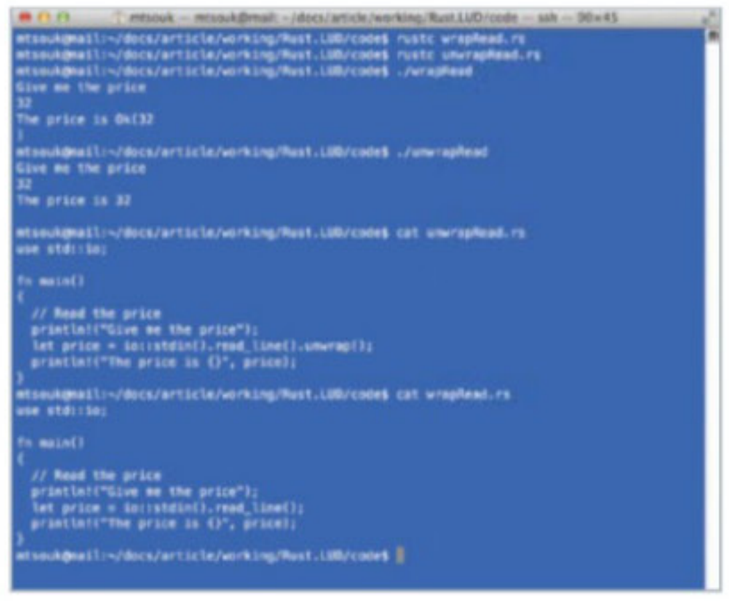
An identifier followed by an exclamation point, such as **println!**, is a macro invocation. Macros are used in Rust because they provide syntactic abstraction. You can find your version of Rust by executing the following command:

```
$ rustc -v
```

Bear in mind that Rust is currently under heavy development, so the Rust version could well be different to the one we're using here.

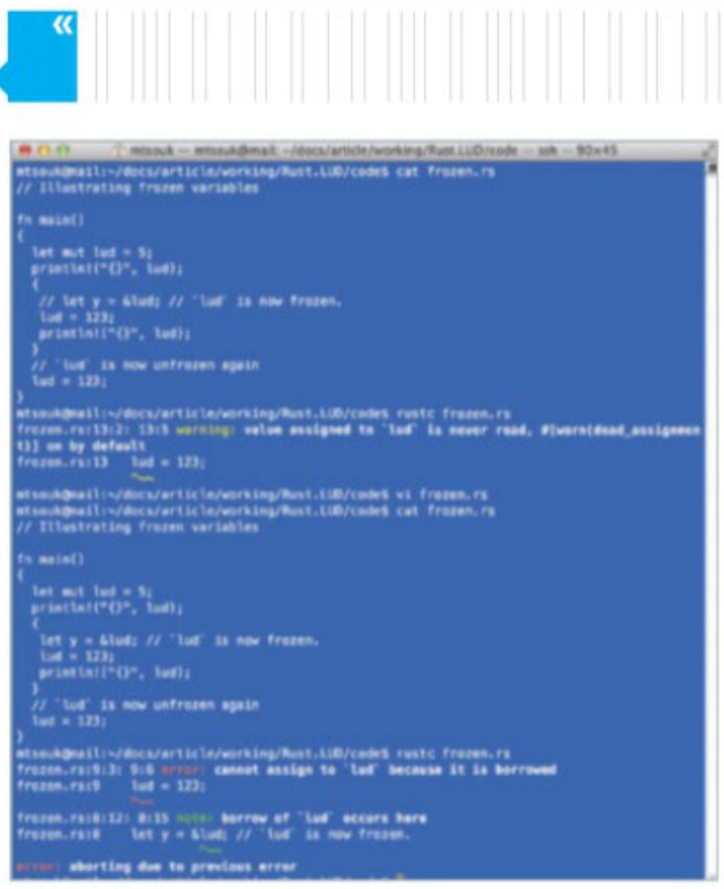
04 Get user input and print output

Printing output can be done with the **println!** macro. This macro is similar in functionality to the **printf** function found in C. Reading a value can be done with the help of the **io::stdin().read_line().unwrap()** function as follows:



```
println!("Give me the price");
let price = io::stdin().read_line().unwrap();
println!("The price is {}", price);
```

It may look a little strange to begin with but you will get used to it soon enough – after all, you are learning a new programming language! The **unwrap()** part of the **io::stdin().read_line().unwrap()** function deletes the OK value that is added in by Rust in the price variable as an indication that it was read without any problems.



05 Pointers, freezing and borrowing

Rust also supports pointers – although you might be happy to learn that you don't need to use them very often. Fully explaining pointers is beyond the scope of this article, but it's good to know that they exist. Although Rust enforces safety while dealing with pointers, it also allows you to break this rule using an unsafe block, implying that you know what are you doing (trying to make your program run faster or because it is required by the task you are trying to implement). You can think of an unsafe block like programming in C using pointers, memory allocation and so on.

Rust also supports freezing: putting an '&' in front of an object freezes it and forbids mutation – even if the object was declared as **mut**. 'Frozen' data cannot be modified via the original object, until all the references to it go out of scope. When data is borrowed, it automatically freezes. Borrowing allows you to access some data, without taking ownership over it. In Rust, resources can only have one owner, who can free or move the memory. The following code illustrates the freezing of data:

```
let mut lud = 5;
{
let y = &lud; // 'lud' is now frozen.
}
// 'lud' is now unfrozen again
```

Trying to modify a frozen variable will cause the compiler to generate an error.

06 Functions in Rust

Functions are declared either at the top level or inside other functions. The **fn** keyword introduces a function. A function has an argument list, which is a parenthesised list of 'name: type' pairs separated by commas. An arrow separates the argument list and the return type of the function. Every autonomous Rust program must have a **main()** function, as it happens in C and C++.

By default, the final expression in the function will be used as its return value, unless a semicolon terminates the expression.

07 Expressions

Rust allows you to use the following unique-to-Rust syntax in order to set the value of the discount variable:

```
let discount =
        if intPrice <= 5 {
            10
        } else if intPrice <= 20 {
            15
        } else {
            25
        };
```

Its main advantages are that it is less error prone as well as being easier to read and understand. Nevertheless, it is not mandatory and you can still use the common syntax instead:

```
let discount;
if intPrice <= 5 {
    discount = 10;
} else if intPrice <= 20 {
    discount = 15;
} else {
    discount = 25;
}
```

08 Implement a function that calculates Fibonacci numbers

The implementation of the function that calculates Fibonacci numbers in Rust uses recursion as usual. The program accepts its input from the command line and uses various Rust features, including access to the command line arguments and vectors. The Fibonacci function is declared as follows:

```
fn fibonacci(n: int) -> int {…€¦
```

This declaration tells that the Fibonacci function takes an integer as its input and then returns an integer as its output.

The access to the vector that holds the command line arguments – skipping the first value because it's the name of the executable – is done using the following for loop:

```
for x in args.iter().skip(1) {…€¦}
```

09 Pattern matching

Rust uses the **match** keyword to do its pattern-matching tasks. match is similar to the switch construct found in C. You provide it with a value and a number of arms – each arm has a pattern – and the code compares the value against each pattern until it finds a match. An example is the following:

```
let database =
  match sqlite::open("lud.db") {
    Ok(db) => db,
    Err(e) => {
      println!("Cannot open lud.db!");
      return;
    }
  };
```

If the **sqlite::open("lud.db")** command returns **Ok(a_database_name)**, then the code in the first arm is executed, giving the **database** variable the value of the **db** variable. Otherwise, an error message is printed (**'Cannot open lud.db!'**) and the code returns. All match constructs must have arms that cover every possible case.

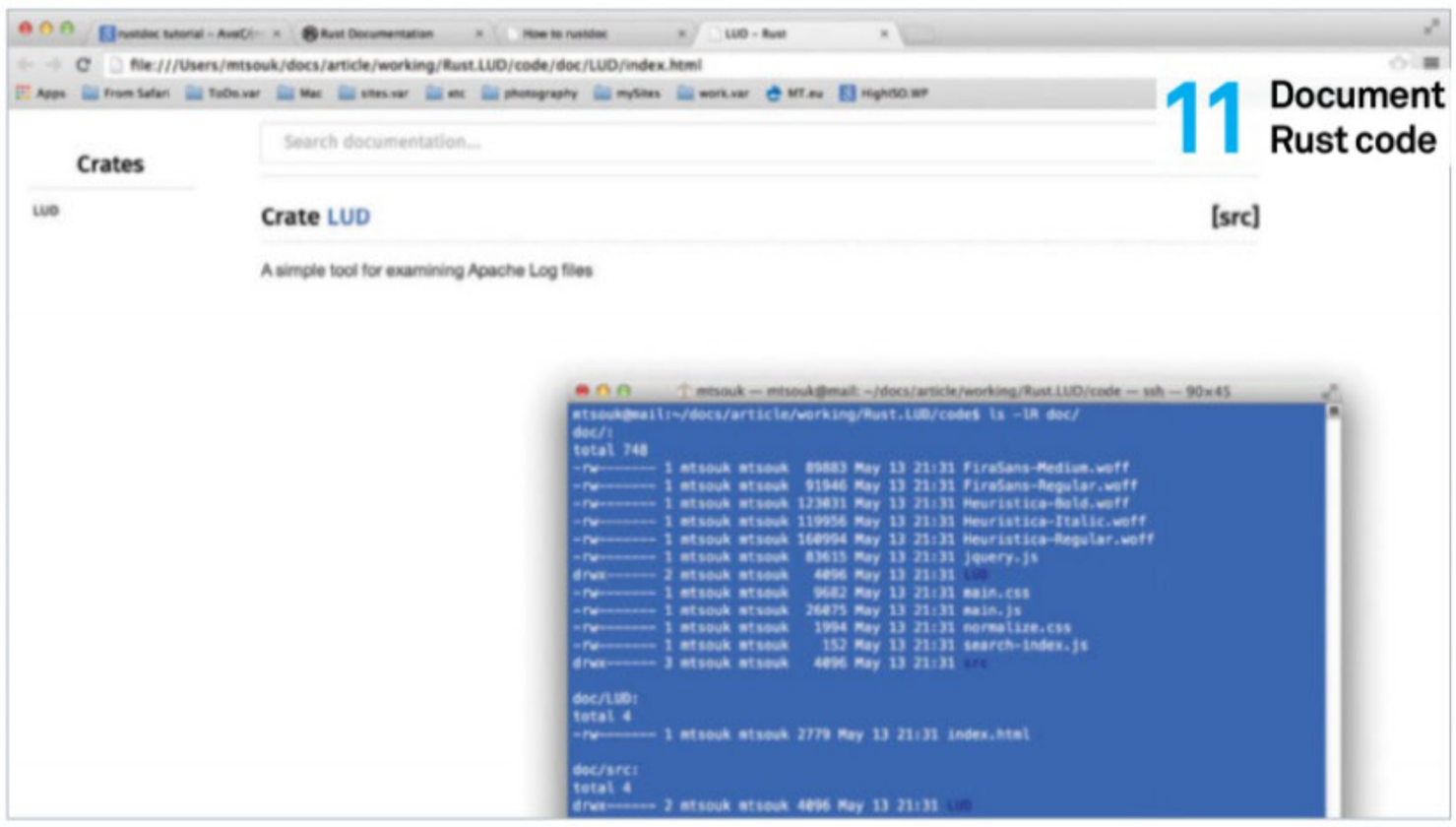
10 Process log files

This part of the article will present a Rust program that demonstrates how Rust deals with reading and writing of files.

Algorithmically speaking, the program will read the log file line by line, while trying to see if the current line contains the desired string and, if yes, it will write the line to a new file. Reading a text file line by line requires the following code:

```
let path = Path::new("logFile.log");
let mut file = BufferedReader::new
(File::open(&path));
for line in file.lines()
{
    print!("{}", line.unwrap());

}
```



"Rust can provide you with elegant HTML out that is more than worth the learning curve"

Meanwhile, creating a new file for writing requires the following code snippet:

```
let lud = Path::new(outputFile);
  let mut fileOutput = match File:: ↵
open_mode(&lud, Open, Write)
  {
    Ok(file) => file,
    Err(e) => fail!("file error: {}", e),
};
```

If you want to write a line to the file, you can do so with the following Rust code:

```
let result = fileOutput.write_
line(myLine);
```

The **fileOutput** file descriptor will be closed automatically on the exit of the scope, so there is no **close()** method.

At this stage, if you choose to run the **ApacheLog** executable you will get output that looks like the following:

```
$ ./ApacheLogs
Line 3 Found! Some(12)
Total lines processed: 5
Total lines written: 1
```

11 Document Rust code

Rust comes with a command line tool called **rustdoc** that helps you generate documentation, provided that you include some documentation-related lines inside your code.

You will have to add the following lines before the use statements of 'ApacheLogs.rs' to make **rustdoc** generate meaningful output:

```
#![crate_id = "LUD"]
#![crate_type="Logs"]

//! A simple tool for examining
//  Apache Log files
```

In order to generate the HTML documentation for the modified ApacheLogs.rs file, you just need to execute the following command:

```
$ rustdoc ApacheLogs.rs
```

Using **rustdoc** does require some practice – as with learning any language – nevertheless, it can provide you with elegant HTML output that is more than worth the learning curve.

Have you got unwanted monitors?

4,000 monitors needed for schools in Africa and Latin America

- WEEE compliance guaranteed
- 0% to landfill policy
- All legal liabilities covered
- Data wiped to UK military standards on all PCs

www.computeraid.org/donate
020 8361 5540
enquiries@computeraid.org

Or find us on...

Computer Aid International

THE ESSENTIAL GUIDE FOR **CODERS & MAKERS**

PRACTICAL Raspberry Pi

66 "The Raspberry Pi is designed to promote an interest in computing – but it doesn't have to be straight-laced"

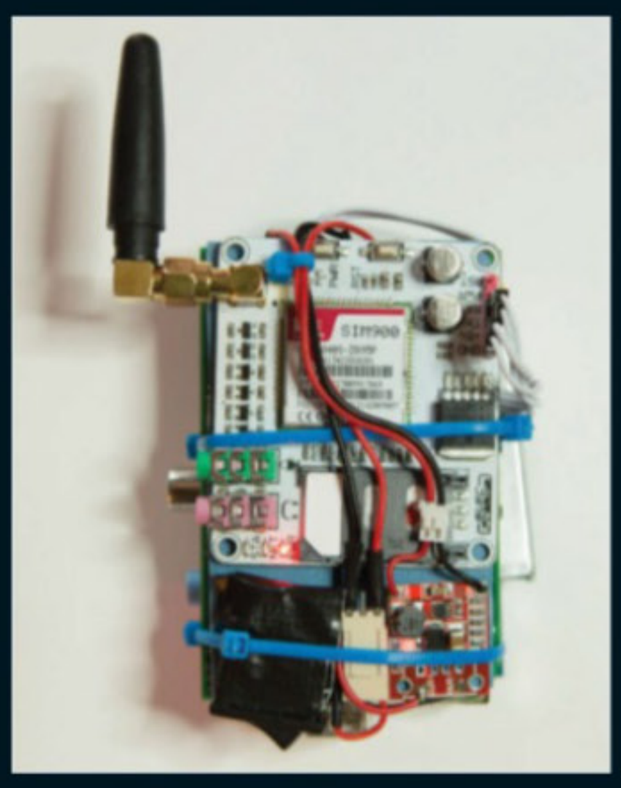
Contents

Touchscreen interface
A simple interface for the touchscreen to dial and make calls, which can easily be improved upon

Hot to handle Powering the entire thing is a Raspberry Pi, which gets a little hot after extended periods of standby time

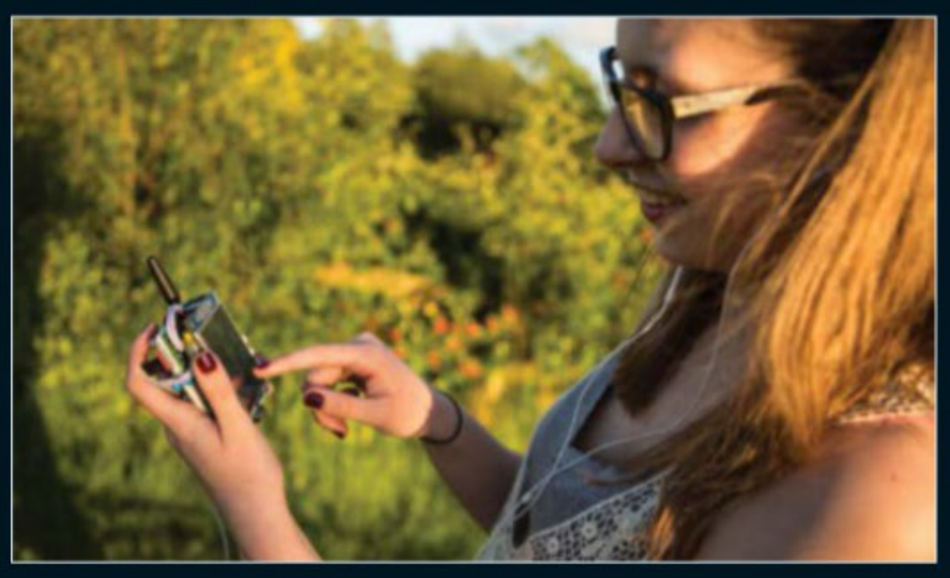
Components list

- Raspberry Pi
- PiTFT Touchscreen
- 2500mAh LiPo battery
- SIM900 GSM/GPRS module
- DC-DC boost converter 3.3V – 5V 1A



Left The GSM module is attached to the back with cable ties – it's functional but not pretty

Below The phone can currently only make calls but the code is open for modification



PiPhone

Turning your Raspberry Pi into a mobile phone is a lot simpler than you'd think, albeit a little chunky...

How did the idea come about for the PiPhone?

I did a presentation at a local conference a couple of weekends ago and I had to think of a topic. So I thought of the evolution of the PiPhone, where it came from.

It started with the Raspberry Pi; I had a project called the Camera Pi, which was embedding a Raspberry Pi into a battery grip of an SLR camera. That went viral; it was nuts because Engadget and everywhere covered this. I'd been waiting to do that project but I couldn't do it until the Pi came out because everything else was either too expensive or too big. The Gumstix were around, and they were nice and small, but still over $200 at the time, then the Raspberry Pi came along and changed everything. So I started making with it and that got a good reaction, and I went on and started making time-lapse rails and bark-activated door openers and macro rails and all sorts of other stuff based on Raspberry Pis.

We really are living in a golden age for kids in terms of electronics: you can buy kits now that we never had – that I never had – as a kid. You can buy GPS units and GSM modules and pressure barometers and all these kits are freely available off Adafruit and via these DIY kits. It's just a fantastic time to be living in for technology and it's great that the Raspberry Pi is building on that and encouraging kids to get these bits and pieces and put them together. Which is what I'm doing with the PiPhone: it's all just off-the-shelf parts that you can put together in a way that hadn't really been done with a Raspberry Pi. It's been done with Arduino but it never really caught on – maybe it's the name that made the PiPhone go viral? Probably!

Have you seen anyone else make a PiPhone yet?

The bunch of people that have said that they're building PiPhones is amazing, there are at least half a dozen people that are actually getting parts and I don't know how many other people are thinking about it and doing it without letting people know. That's the whole idea: get people building and get people playing with Linux even if they might not be that experienced with electronics. The whole PiPhone user interface is written in 150 lines of Python – it's something that you can get your head around.

How usable is it as an actual phone in its current state?

Not very! It basically makes calls but it doesn't receive calls. It makes calls pretty well; I've got the interface so it changes the green dial button to red when it's making a call and when you hang up it goes back to green again. It tells you when it's making a call and it establishes the call, and when you've got your headphones in you can talk and the red button will hang up the call. That part of it works quite well. Functionality-wise it's only a few extra lines of Python to poll the GSM module for calling or listening for a string and then doing an ATA to answer the call. There's nothing at all stopping anyone from adding the extra ten lines of Python to add an answering functionality into it. I'm just waiting to see who's going to do that first – is it going to be one of the people building it? Are they going to extend it a little bit? That would be fantastic.

As for other functionality, there's very little in there. Bearing in mind it's a full Linux/X Windows system behind that Python interface.

What projects do you have planned with the Raspberry Pi in the future?

I've already moved on really. I've just got myself a 3D printer and I'm ramping up on that; I've already started a project that deals with parts and motors and Raspberry Pi's driving motors and such, mounted on 3D prints in interesting ways.

I've done a time-lapse rail and I always wanted to add a panning head, so as well as moving sideways along a rail I always wanted the camera to be able to pan left, right, up and down in a rotational manner. I'm kind of working on that at the moment with parts printed on the 3D printer. It makes a lot of the tools in my shed totally obsolete – files and saws and all sorts of stuff that I would have had to use for making very complicated pieces. Now you just 3D print the part, mount a motor to it and it's there.

So you should see lots more of that coming out but it's still going to be all Raspberry Pi/Arduino based; there's always going to be an embedded computer in it some way.

> It's just off-the-shelf parts that you can put together in a way that hadn't been done before with a Pi

Dave Hunt is an embedded systems engineer that has been making since the early Noughties, currently specialising in Raspberry Pi, Arduino and photography

Like it?

If you too are interested in wireless networking and Raspberry Pis, check out the Onion Pi project from Adafruit, where you can build a secure router: **learn.adafruit.com/onion-pi**

Further reading

Visit Dave Hunt's blog to learn more about the PiPhone, with information on how to build it and links to the code: **bit.ly/1jNLemR**

Raspberry Pi motion detection

Rob Zwetsloot models complex systems and is a web developer proficient in Python, Django and PHP. He loves to experiment with computing

Use the Raspberry Pi camera to detect motion and take a photo of the cause for home security or time-lapse photography

What you'll need

- SD card with up-to-date version of Raspbian
- picamera Python module
- Raspberry Pi camera module

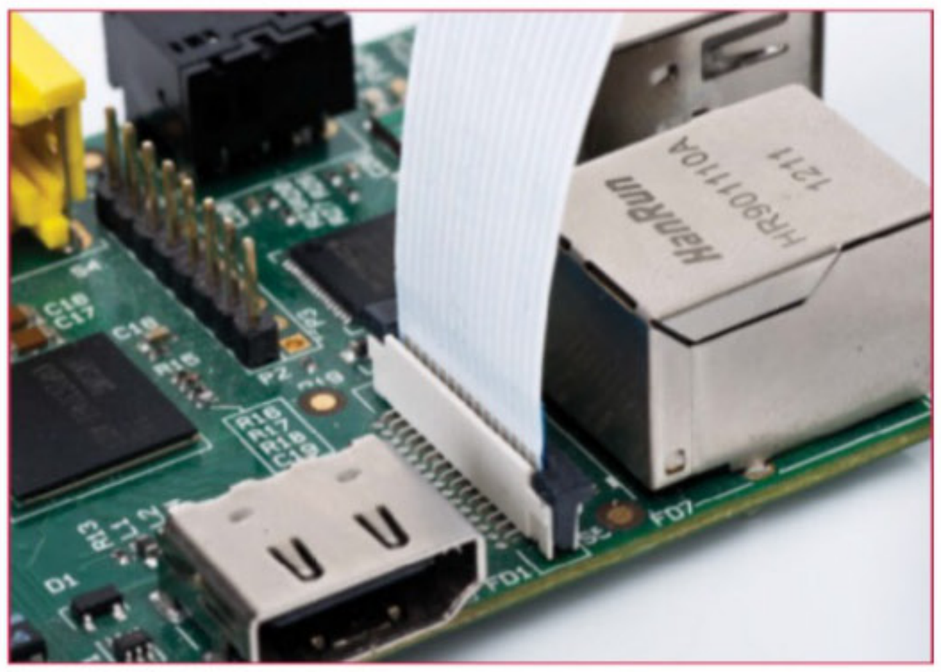
The Raspberry Pi camera is an excellent piece of kit for many projects. With a decent sensor, video capabilities and great Python support from picamera, there's plenty you can do with it. In the past we've done surveillance and time-lapse photography with the camera – but what if you could combine the two? Having the camera take photos when motion occurs reduces the strain on the Raspberry Pi for surveillance and can create a better time-lapse for more human situations. The code is fairly simple as well, plus you'll be able to learn some special methods of storing photos.

Above With modular components, It's easy to disassemble and allows for various configurations to suit your needs

01 Enable the camera module

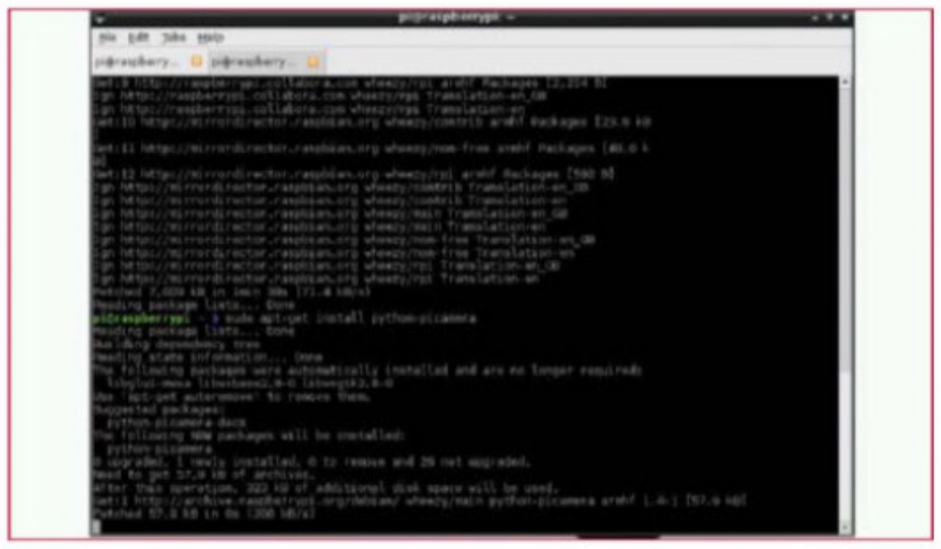
Before we start, make sure to update your Raspberry Pi and its firmware with **apt-get update** and **apt-get upgrade** followed by **rpi-update**. Once that's done, enter the **raspi-config** by typing exactly that in the terminal and choosing 'Enable' in the Enable Camera option. Hit Finish to reboot.

Raspberry Pi motion detection



02 Attach the camera

Turn the Raspberry Pi off either during the reboot process or afterwards. Disconnect the power and find the slim port between the ethernet port and the HDMI port. Lift up the fastener and slot in the camera module's ribbon, with the silver side facing towards the HDMI port.



03 Install the Python modules

The code requires a couple of extra Python modules to work. First, the all-important picamera so we can better control the camera, along with the Python Image Library so we can inspect the images. Install them both with:

```
$ sudo apt-get install python-picamera python-imaging-tk
```

04 Position your camera

Where are you placing your camera? Does the Raspberry Pi have adequate access to power? Are you controlling it with SSH or will it be in range of a display and keyboard/mouse? Set up your Raspberry Pi accordingly and take a sample picture to make sure it's in the right location with:

```
$ raspistill -o test.jpg
```

05 Tweak the sensitivity

In our code, difference and pixels are used to determine when a picture should be taken. The difference variable is the amount a pixel needs to change in colour to count as a change, while pixels is the number of these changed pixels that will be used to determine whether or not enough motion has been created to take a picture.

06 Cross the stream

To make the comparison, we're taking a sample image and inputting it into a stream we've created using **io.BytesIO**. It's stored in memory and then compared with the previous image that has been saved later on in the code – this determines whether or not a new photo should be taken.

Having the camera take photos when motion occurs reduces the strain on the Raspberry Pi

Full code listing

```
import io
import os
import picamera
import time
from datetime import datetime
from PIL import Image

camera = picamera.PiCamera()

difference = 20
pixels = 100

width = 1280
height = 960

def compare():
    camera.resolution = (100, 75)
    stream = io.BytesIO()
    camera.capture(stream, format = 'bmp')
    stream.seek(0)
    im = Image.open(stream)
    buffer = im.load()
    stream.close()
    return im, buffer

def newimage(width, height):
    time = datetime.now()
    filename = "motion-%04d%02d%02d-%02d%02d%02d.jpg" % (time.
year, time.month, time.day, time.hour, time.minute, time.second)
    camera.resolution = (width, height)
    camera.capture(filename)
    print "Captured %s" % filename

image1, buffer1 = compare()

timestamp = time.time()

while (True):

    image2, buffer2 = compare()

changedpixels = 0
for x in xrange(0, 100):
    for y in xrange(0, 75):
        pixdiff = abs(buffer1[x,y][1] - buffer2[x,y][1])
        if pixdiff > difference:
            changedpixels += 1

if changedpixels > pixels:
    timestamp = time.time()
    newimage(width, height)

image1 = image2
buffer1 = buffer2
```

Miles Hodkinson
As a promoter of the Internet of Things, Miles is waiting for the day when all types of people can build their very own custom products

Make a tweeting wireless flood sensor

Flood-proof your basement in just 19 lines of code, or easily tweak the project to create your own personalised alarm system...

Flooding saw hundreds of homes right across the world underwater this year, and many would have benefited from having just that little bit extra warning. In order to be better prepared for floods, we're going to show you how you can prototype your own wireless flood sensor in less than ten minutes. Building it might give you just enough warning to dash home from work, move valuable items upstairs and take the lawnmower, caravan and motorbike to higher ground. Handily, it can also be used to detect toilet flushes, water butt levels or any liquid level rise or fall at all – so it's not just something fun to try out, it's practical too!

Sending tweets

Sending a tweet used to be really easy, if a little on the insecure side. These days you need to register an application with your Twitter account – you do have one, don't you? If not, go create one at **www.twitter.com**. At first this project can look a little daunting, however it can be done painlessly in less than five minutes, if you follow these steps closely!

What you'll need

- **Ciseco** Raspberry Pi Wireless Inventors Kit
  shop.ciseco.co.uk/raswik
- Float sensor
  shop.ciseco.co.uk/float-switch
- DC power supply between 6v and 12v

01 Link Twitter to mobile

Make sure your Twitter account has a mobile phone number associated with it. In your main Twitter account, click the gears icon at the top-right and then 'Mobile' in the list. At this stage, just follow the instructions on screen.

02 Set it all up

With your Twitter username and password, sign in to **https://apps.twitter.com** and click on the button 'Create an application'. In the name field we suggest you use your Twitter account name, add a space and then write 'test'. For the description, just put 'Test poster for Python'. Finally, for the website you can put anything you like. For example, http://www.mywebsite.com – but it's important you don't forget the 'http://'.

03 Enable reading and writing

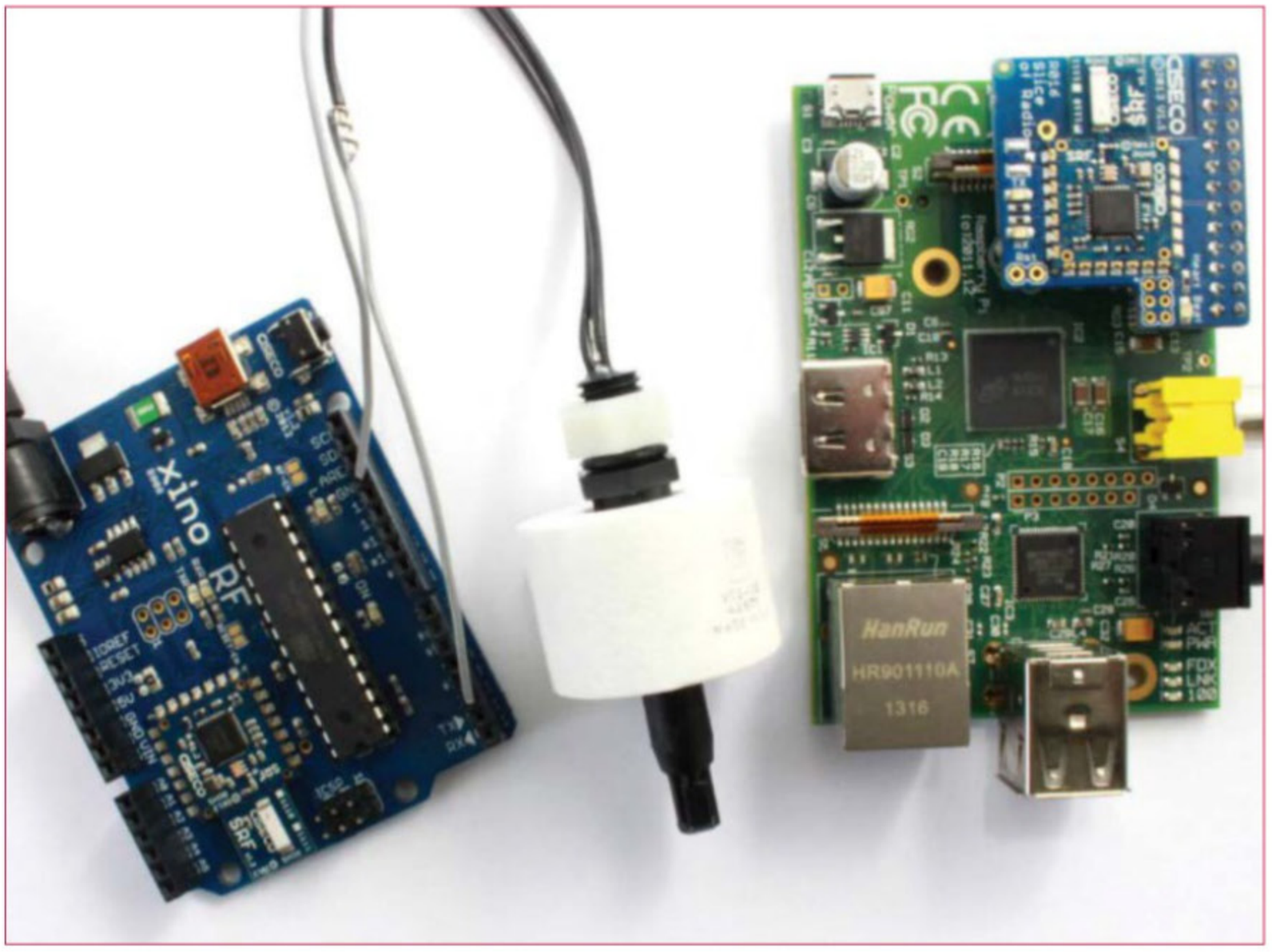
Since you want to be able to send tweets, click on the 'Settings' tab, change to 'Read and Write' and then click 'Update'. This might take a few seconds.

04 Generate codes

Now go back to the 'Details' tab. You will see that an 'API key' and 'API secret' are visible, and that there's a 'Create my access token' button. Click that button to obtain all four of the codes you'll need. If you did this before Step 2, or it still says 'Read', all you have to do is click the button to recreate these codes. It really is straightforward.

05 Remember the codes

Earlier on 'API' was called 'consumer', and you might have come across this before in examples on the web. We suggest copying the following essentials into Notepad so they don't get lost: API key, API secret, Access token and the Access token secret.



Right The Wireless Inventors Kit enables you to connect a Raspberry Pi to an Arduino module from the other side of your house

Make a tweeting wireless flood sensor

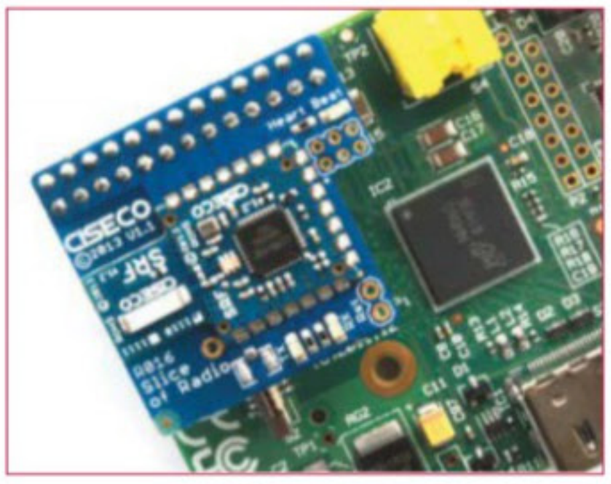
Tweepy is an easy-to-use Python library that works great for accessing the Twitter API

No RasWIK?

Not to worry, using different hardware is always a possibility when playing around with the Raspberry Pi. The reason we chose to use the RasWIK is simply because everything except the float switch is in the box and preloaded with software, making it much easier to get up and running quickly. As a bonus addition, this software is also available to download for free.

To build this with a conventional Arduino or clone, you'll need a USB cable and to leave it 'wired', or use serial-capable radio modules such as the Xbee or APC220. We are, after all, only sending and receiving plain text.

The Arduino sketch used can be downloaded from **http://github.com/CisecoPlc/LLAPSerial**, while the SD image for the OS we used is based on a stock version of Wheezy, which can be downloaded from **http://bit.ly/SfhLLI**.



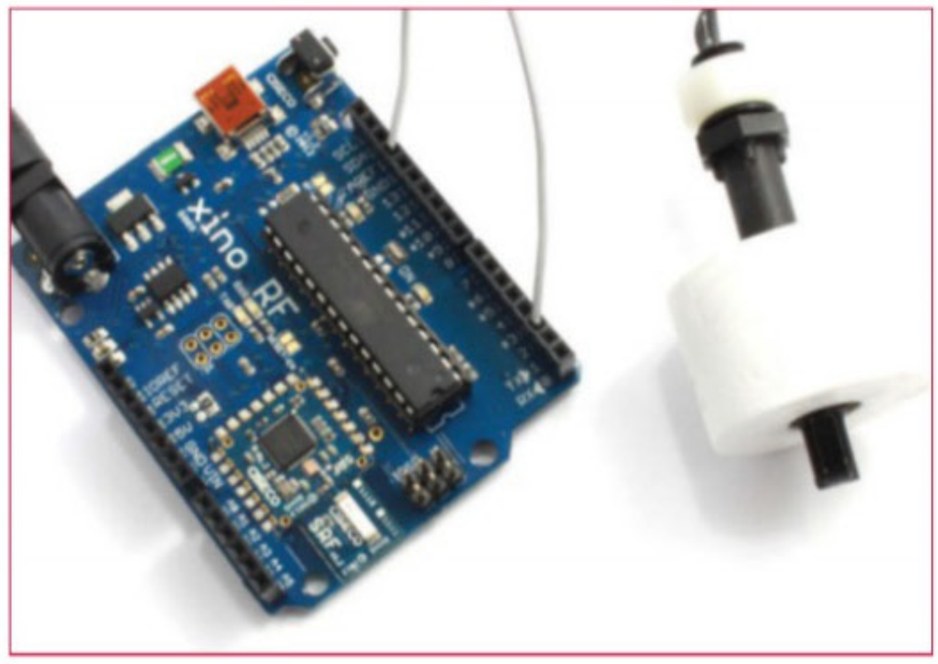
01 Start simple

To get going with your flood sensor, simply plug in the Slice of Radio to your Pi and insert the preconfigured Raspbian operating system.

02 Go to LX terminal

Power up the Raspberry Pi, log in and type **STARTX** to start the desktop. Double-click the LX Terminal and type the following into the black window:

```
minicom -D /dev/ttyAMA0 -b 9600
```



03 Make the connection

Connect the float switch to the XinoRF ground pin (marked GND) and digital I/O pin2. Then, power up the XinoRF (you will see a--STARTED– displayed in minicom)

04 Test the sensor sends messages

Wiggle the sensor up and down (you should get a--D02HIGH– when the sensor position is up) and see the RXR LED on the XinoRF flicker with each message sent.

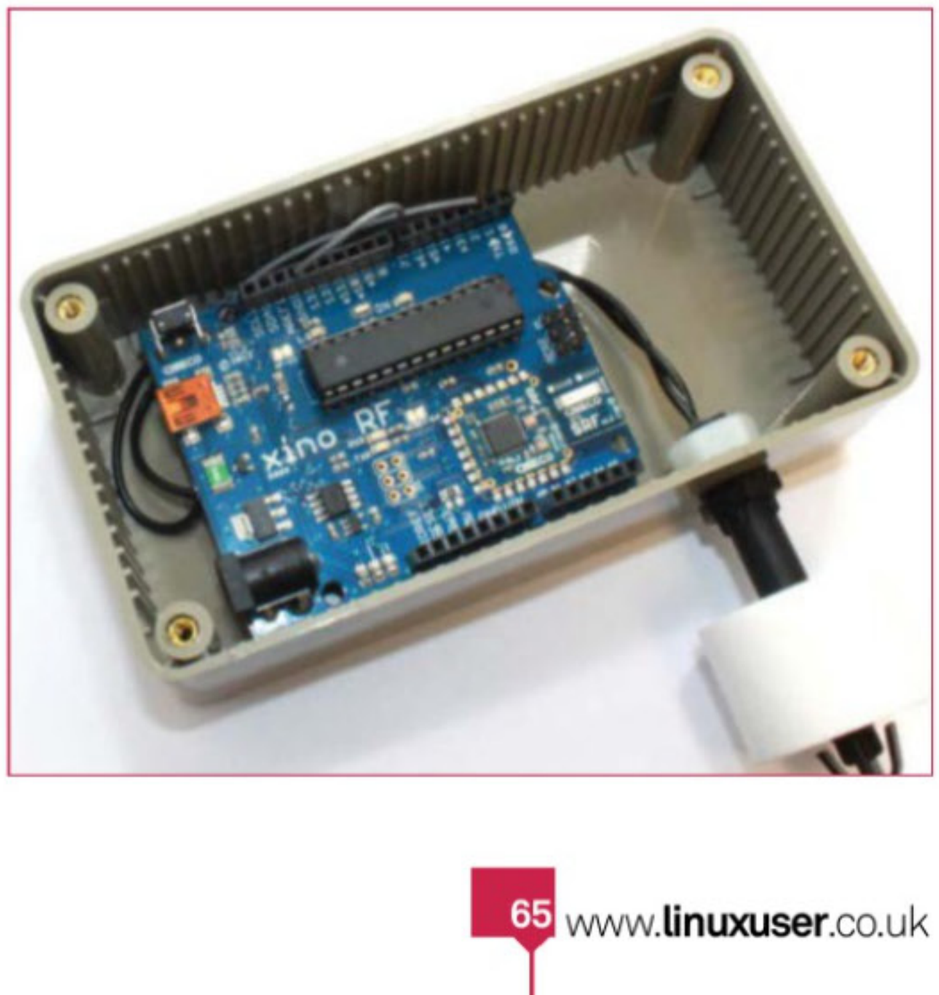
05 Install Tweepy

Tweepy is an easy-to-use Python library that works great for accessing the Twitter API. For more information or to check out the documentation, visit **https://pypi.python.org/pypi/tweepy/2.2**. Type in a shell window the following:

```
sudo pip install tweepy
```

06 Put the sensor to work

Test your prototype using a regular saucepan of water. If you want to put your flood sensor to real use then place it into a waterproof box and ensure it is mounted securely.



Full code listing

```
#  Let's set up how to talk to Twitter first

import tweepy, serial, time
API_key = "GWip8DF9yf0RYzjIlM6zUg"
API_secret = "Yjipapo56nhkS2SAWtx3M4PAit3HsvWZUYAOghcLn4"
Access_token = "2392807040-19lSoaVOmj8NTvJVteU8x265IPEw2GfY0cS7vuN"
Access_token_secret = "lV4u2ls4oeRZCAxcpaPQNzRDN0lSiUibY0MdCuYKk16Rl"
auth = tweepy.OAuthHandler(API_key, API_secret)
auth.set_access_token(Access_token, Access_token_secret)
api = tweepy.API(auth)

#  Open a com port (yours may differ slightly)

ser = serial.Serial('COM3', 9600)

#  An endless loop checking for sensor alerts

while True:
    SerialInput = ser.read(12)
    if SerialInput == 'a--D02HIGH--':
        TimeNow = datetime.datetime.now()
        DS = TimeNow.strftime('%H:%M:%S %d-%b-%Y')
        AlertText = 'ALERT: LLAP+ device -- flood sensor triggered @ ' + DS
        print (AlertText)
        api.update_status(AlertText)
        time.sleep(10) #stop fast re-triggering
        ser.flushInput()
```

Leo White
develops software for embedded Linux devices in his day job and tinkers with Raspberry Pis and robots in his free time. More ramblings can be found at **bit.ly/1l8wBPk**

Build a Raspberry Pi-powered Bigtrak

Take a toy, a Raspberry Pi and a PS3 controller; add a dash of Python and some solder for the perfect remote-controlled gadget...

What you'll need

- Bigtrak
  www.bigtrakxtr.co.uk/shop/bigtrak
- Breadboard and cables
- Motor driver bit.ly/1i0nFug
- USB Battery pack
  amzn.to/1h2PBil
- PS3 DualShock controller

Build a Raspberry Pi-powered Bigtrak

The Raspberry Pi is a small, low-cost computer designed to promote an interest in computing and programming – but it doesn't have to be straight-laced computing. In fact, in this article we'll be showing you how you can use it to turn a Bigtrak into a robot. That's educational, right?

The Bigtrak is a toy that takes in a list of straightforward commands (Go forwards, turn left, turn right) and then executes them. To make things more interesting we're going to remove the existing circuitry and replace it with a Raspberry Pi, using a small motor driver to safely control the motors in the Bigtrak, which we'll then set up to be controlled via a PlayStation 3 DualShock controller.

Everything required on the software side comes pre-installed on the latest Raspbian OS images, so all we need to translate changes from the controller to the motors is a small Python script that uses the Pygame and RPI.GPIO modules.

01 Opening up the Bigtrak – the easy bit

Before we can make any changes to the Bigtrak we need to get inside. First, flip the Bigtrak upside down and remove the nine screws from around the edge. These are mostly easy to get at, however the ones on the front may require a more slender screwdriver to reach them.

02 Opening up the Bigtrak – the fiddly bit

The last two screws are located underneath the grey grille on the back. This grille is held in place by four plastic tabs that need to be pushed in while at the same time sliding the grille away from the Bigtrak. This can be quite tricky as there is limited space to get extra hands in to help out. It can help to wedge some thin plastic items (eg a guitar pick) into the sides to keep those two tabs unlocked, while using your fingers to push in the bottom two tabs and slide the grille upwards, allowing you to remove the screws.

03 Removing the top

Put the Bigtrak back onto its wheels then carefully loosen the top and lift upwards. The lid is connected to the base with a ribbon cable and a switch, so only pull the top up far enough for you to tilt it to one side and expose the inside.

With the lid lifted up onto one edge, remove the screw holding the switch in place and detach it from the lid. Next, you need to unscrew the two screws on the PCB that hold the ribbon cable in place and let it slip free.

With the switch and ribbon cable disconnected, the lid should now come free and can finally be completely removed from the base of the Bigtrak.

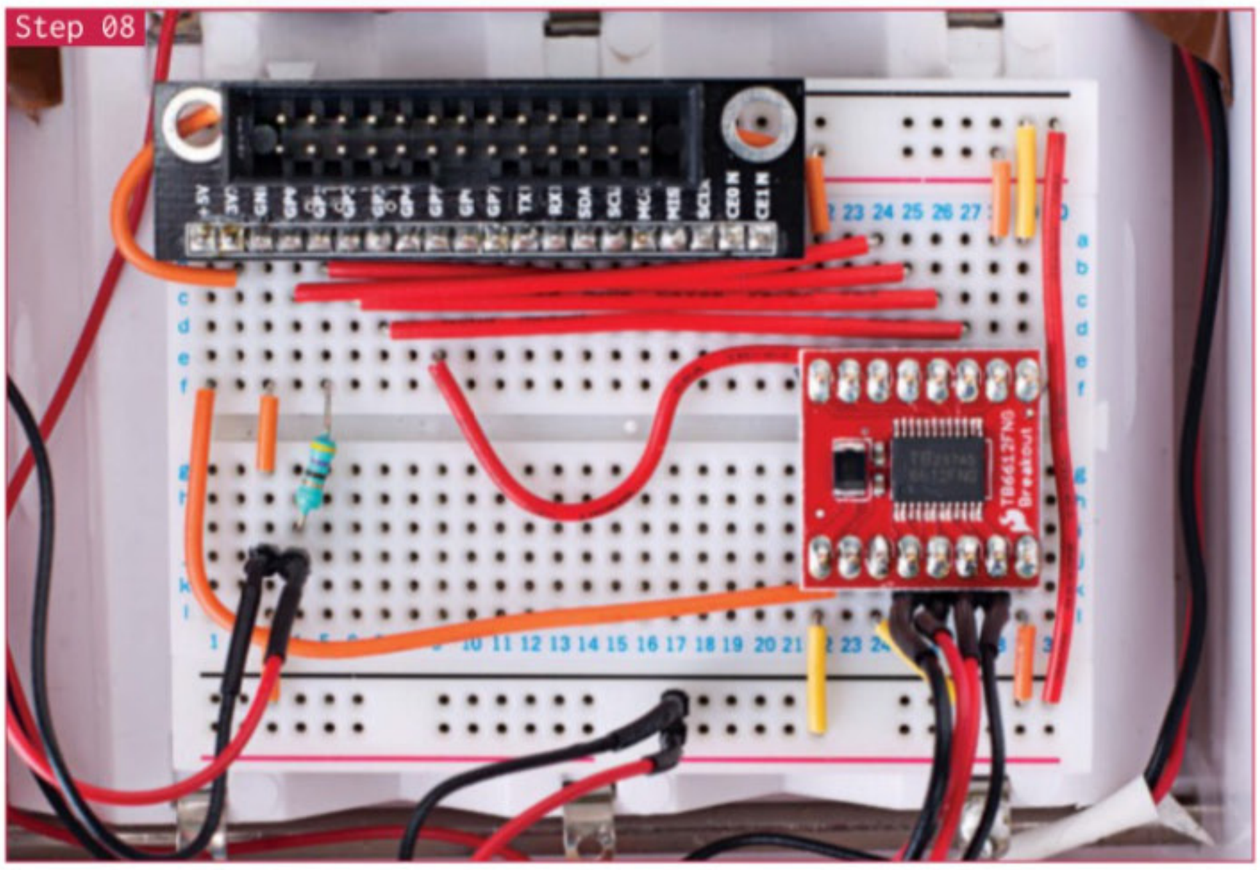
04 Cut the wires

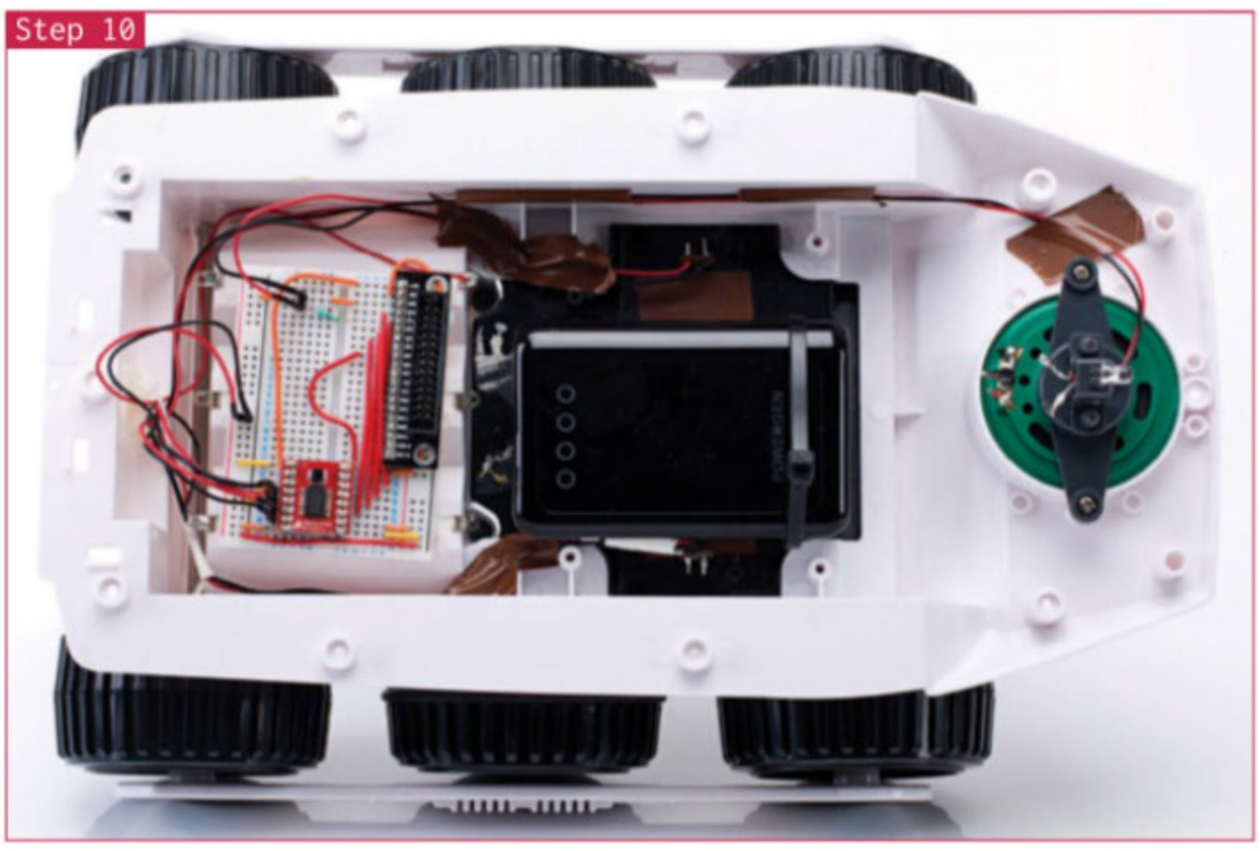
Cut the wires leading to the main PCB. The ones for the switch and power should be cut close to the PCB (so we can reuse them later) whereas the ones to the LED and speaker can be cut wherever you like.

05 Remove the engine

Turn the Bigtrak upside down and remove the four screws holding the engine in place (this will reduce the chance of soldering iron damage to the main body). Carefully turn the Bigtrak back over and lift it up until the engine slips free.

The wires need to be long enough to reach the back of the Bigtrak, so be generous!





06 Rewire the motor

Remove the solder connecting the PCB to the motors (a solder mop is useful here) and then remove the PCB. With the PCB removed we can now attach wires to the motors in order to drive them from the Raspberry Pi, as opposed to the on-body commands. The wires will need to be long enough to reach the back of the Bigtrak, so be generous – after all, it's far easier to trim long wires to length than replace short wires entirely!

Having installed all of the wires, you can now replace the engine back into the Bigtrak.

07 Connect the motor driver

With the motors back in place we now need to build up a circuit to drive it from the Raspberry Pi. We've used a ribbon cable to connect the GPIO pins on the Raspberry Pi to a breadboard, before connecting it up to a Dual Motor Driver (**http://proto-pic.co.uk/motor-driver-1a-dual-tb6612fng**) to actually drive the motors. This keeps the higher voltage the motors require away from the sensitive GPIO pins.

The connections made on the breadboard are listed in the table below. These values will be needed when writing the software and may be different depending on the breakout board you are using, and the Raspberry Pi revision.

RPi GIPO	Motor Driver
24	AIN2
17	AIN1
18	STBY
21	BIN1
22	BIN2

With the PWMA and PWMB pins directly connected to the 3.3V power rail, the motors will now always run at full speed for as long as they're active.

08 Install the breadboard

The breadboard is going to be installed on top of the battery compartment inside the Bigtrak, so the wires from the motors should be brought to the back to the unit and cable-tied into place. The wires to the batteries can also be brought back to the same place to help keep things tidy.

09 Wire it all together

In order to easily connect the motors and batteries to the breadboard we have soldered some modular connector plugs to the ends of the cable, allowing them to just push into place (these are available from **www.maplin.co.uk/modular-connectors-5348**).

With the breadboard installed (sticking it into place for support) we can now, after double-checking all the connections, plug the motors and power into it. To know when the motors are enabled (and reduce the chance of unexpected movement), the LED can be connected to the breadboard so that it lights up whenever the 'standby' pin is active, using a resistor to ensure it doesn't pull too much current and go 'pop'.

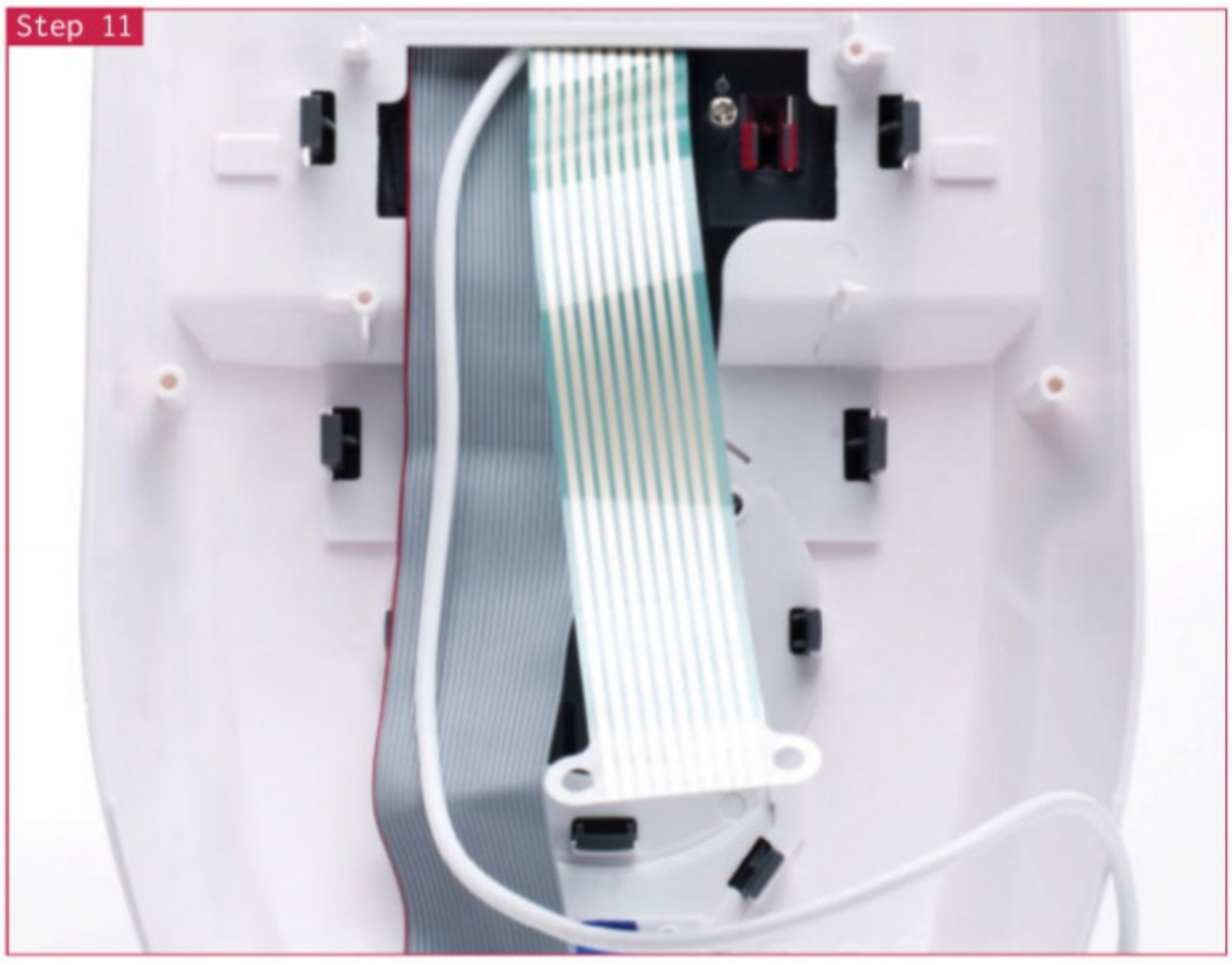
10 Provide power

Power for the Raspberry Pi is supplied via a USB battery pack that is installed on top of the engine and can be held in place by a couple of cable ties or a Velcro strip. This type of battery is typically sold as a portable mobile phone or iPad charger – the one used here is rated at 8000mAh, able to power the Raspberry Pi for up to eight hours.

Build a Raspberry Pi-powered Bigtrak

11 Connect to the Raspberry Pi – adding cables

As the Raspberry Pi will be mounted on the top of the Bigtrak, we need to run the ribbon and power cable through the top of the case. To do this, turn the top of the Bigtrak upside down and release the front two catches that hold the dark grey plastic in place – this provides a big enough gap to feed the ribbon cable and USB power cable through. Make sure that the red edge of the ribbon cable correctly matches up with the connector on the breadboard to save yourself from having to twist the cable inside the case.



12 Connect to the Raspberry Pi – final steps

With the top of the Bigtrak back on, the Raspberry Pi can now be put in place, keeping the GPIO pins towards the front to allow the ribbon cable to easily connect. As for the battery pack, we're holding it in place with cable ties and sticky pads. In theory it's possible to attach the bare Raspberry Pi to the Bigtrak, however this can cause the SD card to press against the edge and bend, so it's recommended to use a case to protect the Raspberry Pi.

Connect the ribbon and power cable to the Raspberry Pi, turn it on and it's now ready to be told what to do. For setting up the software it may be easier to connect up a keyboard and monitor to the Raspberry Pi at this point.

13 Connect the PS3 controller

This should be a simple case of plugging the PS3 controller into one of the USB ports, as all the software to support it is included in the default Raspbian OS image and it will be automatically detected as a joystick. To confirm that the PS3 controller has been detected, run **lsusb** and checked that it appears in the resulting list.

14 Run the software

Now with the system all set up, it should just be a simple case of copying the 'bigtrak.py' file found on this issue's disc onto your Raspberry Pi and running it. As the script accesses the GPIO pins, it will need to be run as the superuser, so launch it using:

```
sudo python bigtrak.py
```

Now we can control the Bigtrak using the analogue sticks! Moving the left stick will control the left motor and moving the right stick will control the right. So, to move forwards push both sticks up, pull both down to go backwards and push one up and one down to rotate on the spot.

If the analogue sticks are not controlling the Bigtrak as expected, double-check the GPIO connections to make sure that they are all as expected.

15 Next steps

Now that you have a solid base for your Raspberry Pi robot, you can make further adjustments to it. Possible next steps could be: add a USB Bluetooth adaptor so the PS3 controller can be connected wirelessly; replace the breadboard with a PiPlate or 'Slice of Pi' add-on board, allowing the Raspberry Pi to be installed inside the Bigtrak; connect up the RaspberryPi camera and a USB WiFi adaptor to stream video as you drive around; or add a robot arm!

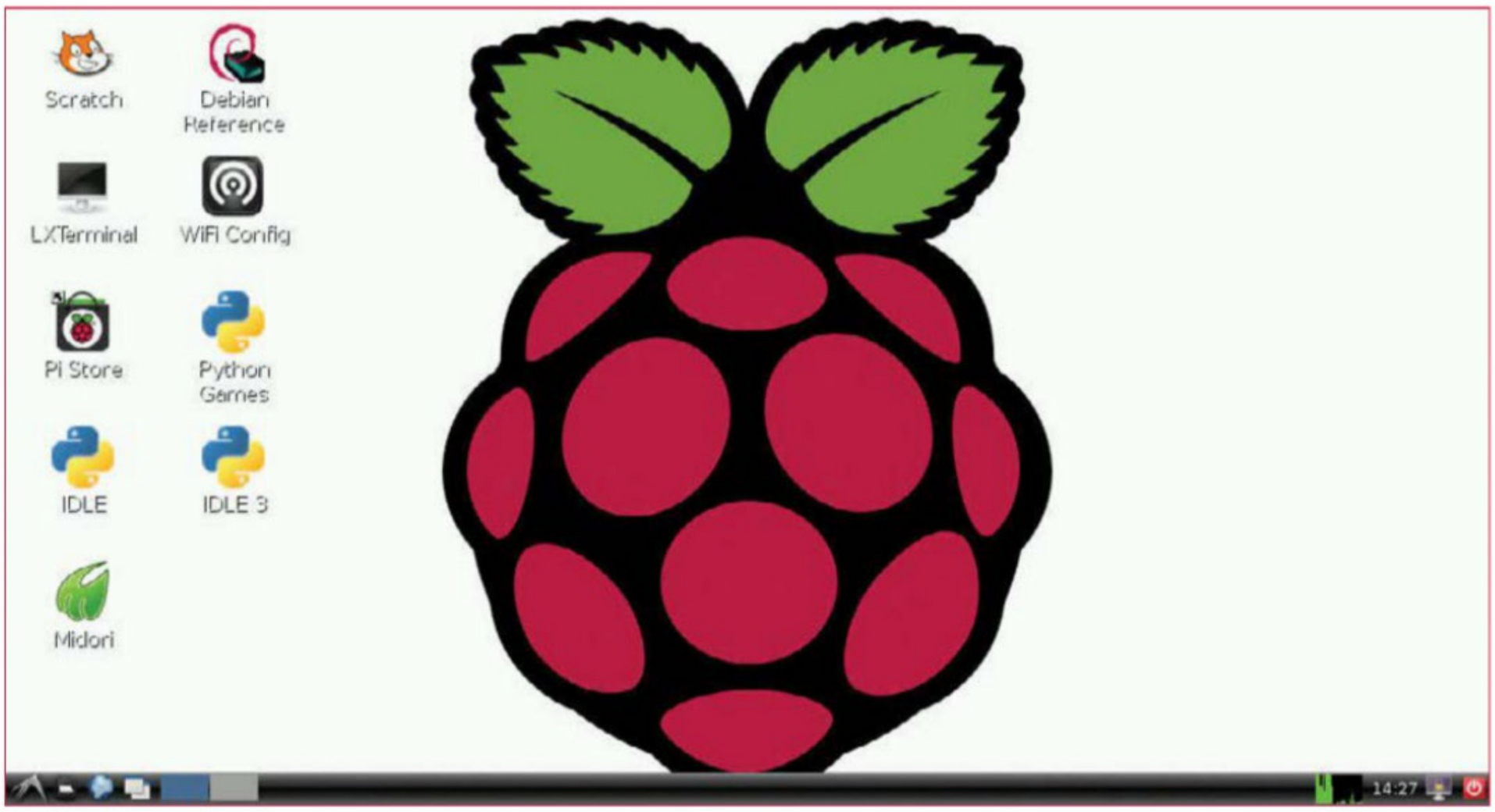
Rob Zwetsloot models complex systems and is a web developer proficient in Python, Django and PHP. He loves to experiment with computing

If you like this...

Check out our brand new **20 Amazing Raspberry Pi Projects** digital bookazine – every project uses Raspbian to make your Pi more than just a computer. More info overleaf.

Further reading

You can grab the latest image from the Raspberry Pi Foundation website but you'll get more information about the distro from its website: **www.raspbian.org**



What is Raspbian?

The operating system that helps to power a Raspberry Pi – but what exactly is a Raspbian and what makes it Linux?

Raspbian! I use it, you guys use it, I don't think there's any tutorial I've done that doesn't involve using it. What exactly is Raspbian, though? It's Linux, correct?

Yes! Raspbian is based on Debian, which is a Linux distribution. It's called Raspbian because it's a portmanteau of Raspberry and Debian; much in the same way as RaspBMC or Pidora.

That answers part of my question, but could you elaborate more on the Debian bit in particular?

Debian is a line of quite popular Linux distributions that is the base of many famous distros; Ubuntu and its offshoots, Tails, SteamOS wattOS and many more. It's very easy to use and contains a large repository of software, so it's easy to extend and customise it exactly how you want it.

You keep calling it a Linux distribution, which seems oddly specific. What does that mean?

The different versions of Linux are called distributions or distros. They take the Linux kernel and its other technologies and package it with their own software, software repositories, branding and perhaps even a desktop environment. Debian is an example of one of these distros.

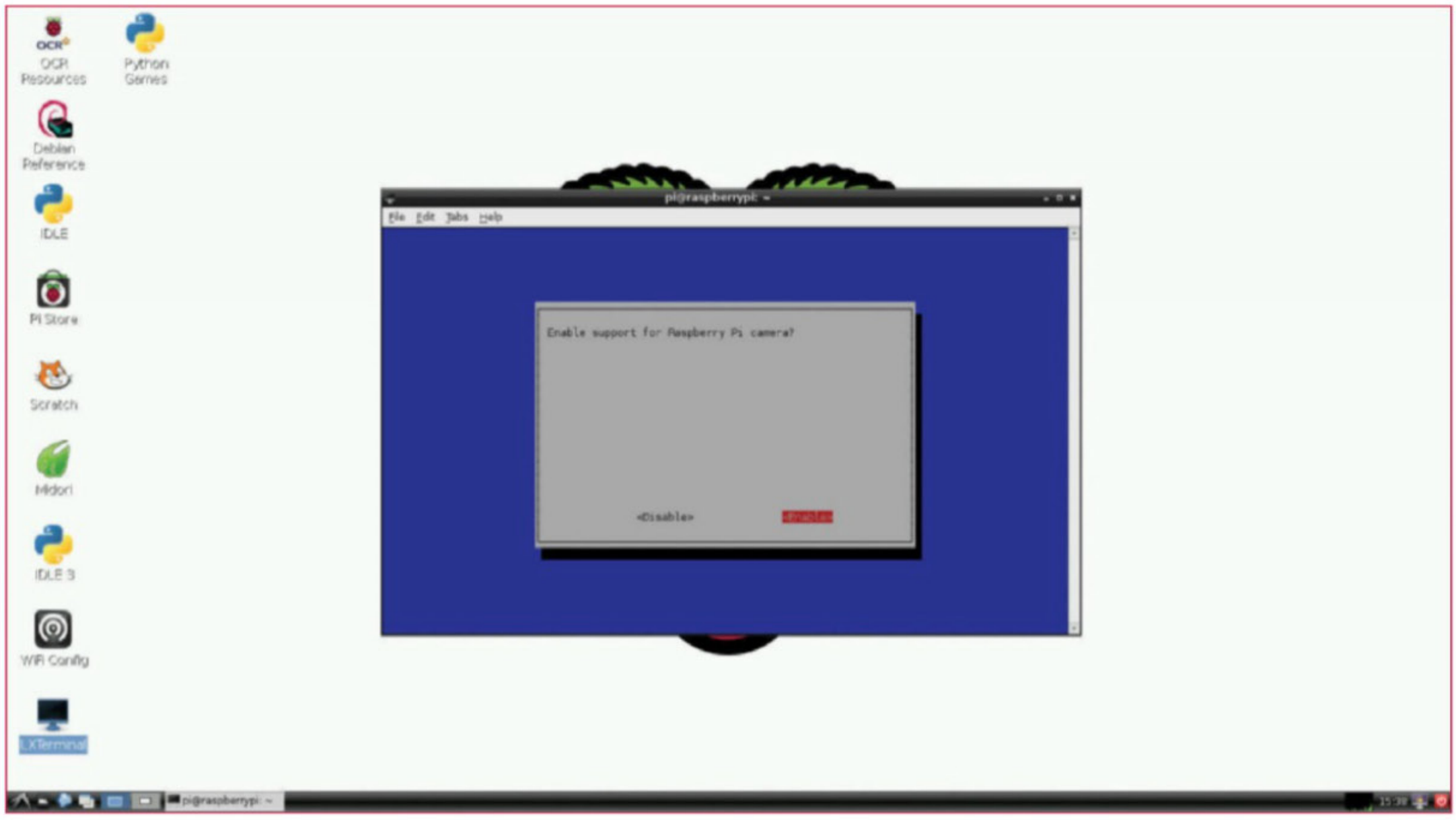
What sort of software makes it stand out as Debian specifically?

There's not a huge amount, to be perfectly honest – and neither is that majorly important unless you're into the really low-level security and behind-the-scenes workings. What you will notice is its package management system, Aptitude, which is a major part of Debian and its various flavours and offshoots – if you've used **sudo apt-get** anywhere in a tutorial, this is it. There's also stuff like the way boot time programs are loaded and some necessary tools for compiling software, but you won't often come across it.

So is Linux an operating system?

That's the $64,000 question. The short answer is no but that doesn't really explain why it's not. Linux is a kernel, which, while a core part of any operating system, does not make up the entire operating system. There are various tools that come with the kernel via the GNU project (the reason why it's sometimes called GNU/Linux) which bring it up to a level more akin to an operating system – but it's not quite there yet. You could technically say that these distributions as a product on their own are part of an operating system – eg Raspbian is an OS for the Raspberry Pi – however, in doing so you run into other issues, such as the stuff making up Raspbian being extra components on top of the Linux kernel and GNU tools which wouldn't make up an operating system on its own either.

If someone asks you if Raspbian is an operating system, though, just say yes. You'll keep more friends that way.



Above Raspbian is the official operating system of the Raspberry Pi, with a simple interface that has access to all the core applications

The project that became Raspbian was chosen as a preferred distro and the community latched onto it

That's a little bit mind-boggling but I think I understand. So if Raspbian is a version of Debian, is there 'pure' Debian on Raspberry Pi?

Yes there is. In fact, there was Debian for Raspberry Pi before there was Raspbian. It was the version specifically designed for the CPU that the Raspberry Pi uses, as not every Linux distro has a version that will run on Raspberry Pi.

Isn't the Raspberry Pi running on ARM? I thought there were other operating systems that ran on ARM?

Yes, there are ARM versions of Ubuntu and even Windows right now, however they're for different versions of ARM. The Raspberry Pi uses ARM v6, while Ubuntu specifically uses a newer version. There are also special tools that make better use of the Raspberry Pi hardware and are integrated into the major Raspberry Pi distros.

So if Raspbian is based on Debian, does that mean there are other distros based on Raspbian?

The Kano OS that we review in this issue (pages 12-13) is based on Raspbian – and there are maybe a couple of others that are minor changes on Raspbian. Still, mostly the other Debian-based Pi distros are actually purely based off of Debian, such as RaspBMC the XBMC Pi distro.

So why is it that Raspbian has become the default distro for the Raspberry Pi, then?

It's mostly down to luck and good timing, really. A couple of years ago there was a Fedora spin that was being touted as the official Raspberry Pi distro, however there were some major problems with it. The project that became Raspbian ended up being chosen as a preferred distro around that time and the community really latched onto it. Since then, all the major Raspberry Pi Foundation announcements regarding software have involved Raspbian and pretty much all shared community projects are done on Raspbian, which is why we do a lot of our projects on it as well.

BUILD A BETTER WEB

www.webdesignermag.co.uk

FREE DISC EVERY ISSUE

web designer

Available from all good newsagents and supermarkets

ON SALE NOW

Supercharge WordPress | A new dimension with Web 3D | Google Maps masterclass

BUY YOUR ISSUE TODAY

Print edition available at www.imagineshop.co.uk

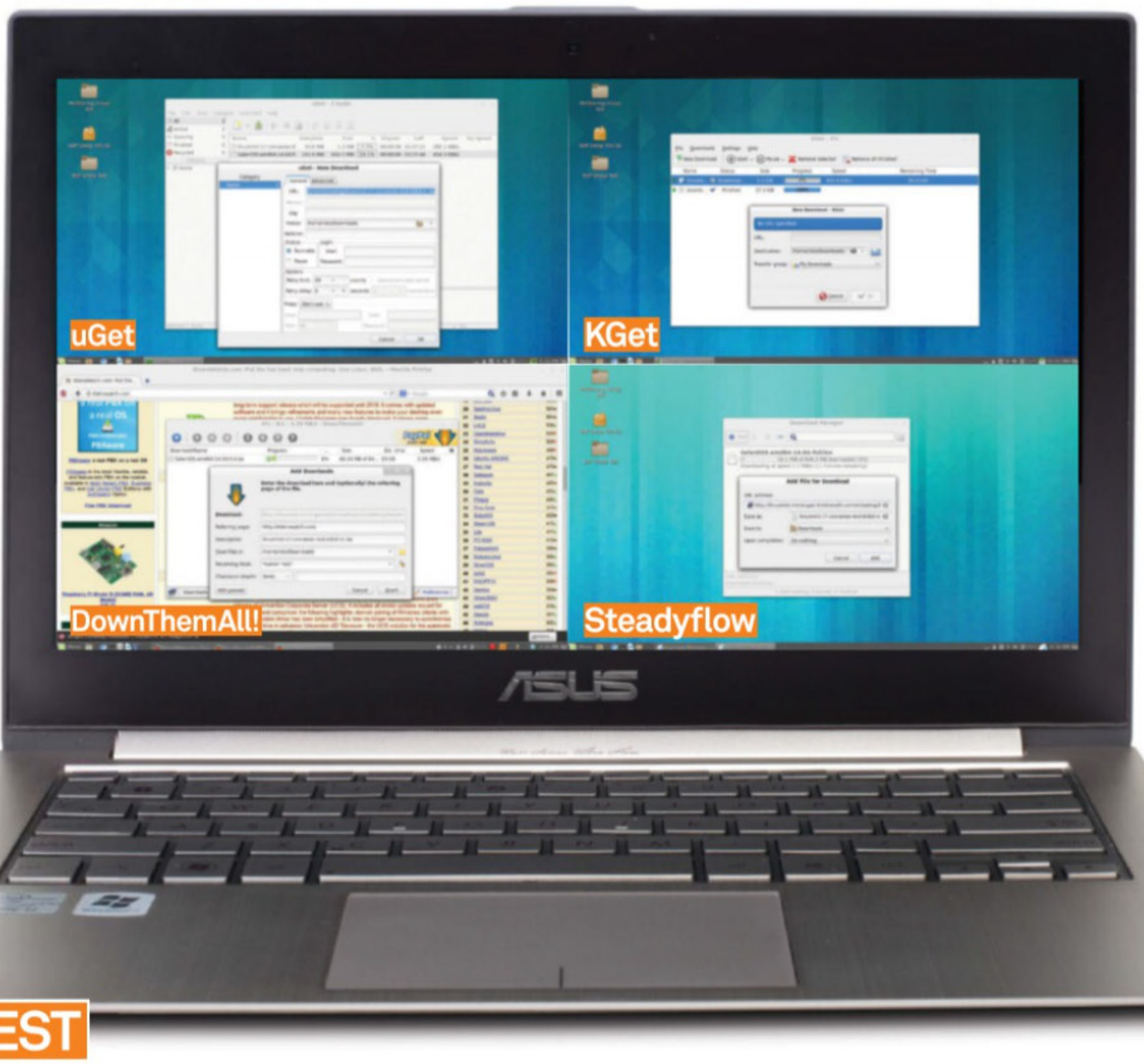
Digital edition available at www.greatdigitalmags.com

Available on the following platforms

Print | Tablet | Phone | Desktop

facebook.com/webdesignermag twitter.com/webdesignermag

Reviews



GROUP TEST

Download managers

Improve and better manage your web downloads for mirroring, mass grabs or just better control over your files

uGet

Advertised as lightweight and full-featured like a majority of other Linux apps, uGet can handle multi-threaded streams, includes filters and can integrate with an undefined selection of web browsers. It's been around for over ten years now, starting out as UrlGet, and can also run on Windows.

Download: bit.ly/1mx4Uwz

KGet

KDE's own download manager seems to have been originally designed to work with Konqueror, the KDE web browser. It comes with the kind of features we're looking for in this test: control of multiple downloads and the ability to run a checksum alongside the downloaded product.

Download: bit.ly/1lilqU9

DownThemAll!

DownThemAll, being somewhat platform-independent, comes to Linux by way of Firefox as an add-on. This limits it somewhat to use with only Firefox, however as one of the most popular browsers in the world its tighter integration may be just what some are looking for in a download manager.

Download: bit.ly/1pWmZdQ

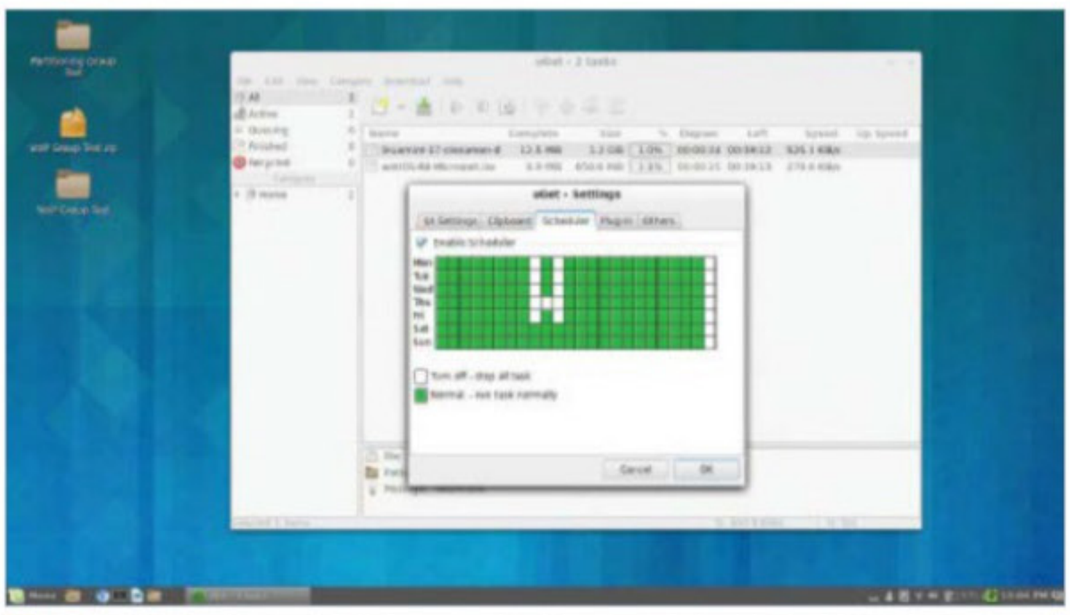
Steadyflow

Easily available in Ubuntu and some Debian-based distros, Steadyflow may be limited in terms of where you can get it but it's got a reputation in some circles as one of the better managers available for any distro. It can read the clipboard for URLs, use GNOME's preset proxies and has many other features.

Download: bit.ly/1lilymS

uGet

Cross-platform, full-featured and lightweight – but are these just claims?



uGet is actually very full-featured, with a lot of the kind of functions that advanced torrent clients use

Interface

uGet reminds us of any number of torrent client interfaces, with categories for Active, Finished, Paused and so on for the different downloads. Although there is a lot of information to take in, it's all presented very cleanly and clearly. The main downloading controls are easy to access, with more advanced ones alongside them.

Integration

While it can see into the clipboard for URLs, uGet doesn't natively integrate into browsers like Chromium and Firefox. Still, there are add-ons for both these browsers that allow them to connect to uGet: Firefox via FlashGot and Chromium with a dedicated plug-in. Not ideal, but good enough.

Features

uGet's maturity affords it a range of features, including advanced scheduling to switch downloading on and off, batch download via the clipboard and the ability to change which file types it looks for in the clipboard. There are plug-in options, but not a huge amount.

Availability

While it's also available in most major distro repos, the uGet website includes regularly updated binaries for a variety of popular distributions as well as easily accessible source code. It runs on GTK 3+ so it has a smaller footprint in some desktop environments than others, although we'd say it's worth the extra dependancies in KDE or other Qt desktops.

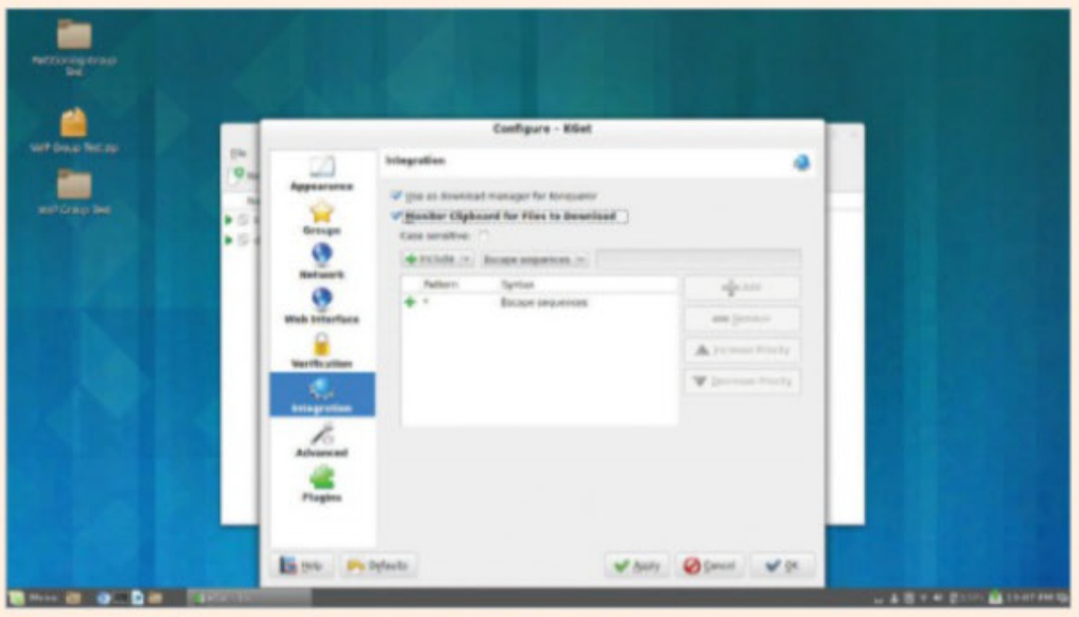
Overall

We very much like uGet – its wide variety of features and popularity have allowed it to develop quite a lot to be an all-encompassing solution to download management, with some decent integration with Linux and browsers.

8

KGet

With KDE behind it, does KGet come out on top?



You need to manually activate the ability to keep an eye on the clipboard for links

Interface

As expected of a KDE app, KGet fits the aesthetic style of the desktop environment with similar icons and curves throughout. It's quite a simple design as well, with only the most necessary functions available on the main toolbars and a minimal view of the current downloads.

Integration

KGet natively integrates with KDE's Konqueror browser, although it's not the most popular. Support for it in Firefox is done via FlashGot as usual, but there's no real way to do it in Chromium. You can turn on a feature that asks if you want to download copied URLs, however it doesn't parse the clipboard very well and sometimes wants to download text.

Features

The selection of features available are not that high. No scheduling, no batch operations and generally an almost bare-minimum amount of downloading features. The clipboard-scanning feature is a nice idea but it's a bit buggy. It's a little weird as the Settings menu looks like it's designed to have more settings and options.

Availability

While it doesn't come by default with a KDE install, it is available for any distro that supports KDE. It does need a few KDE libraries to run though, and it's a bit tricky to find the source code. There isn't a selection of binaries that you can use with a few distros either.

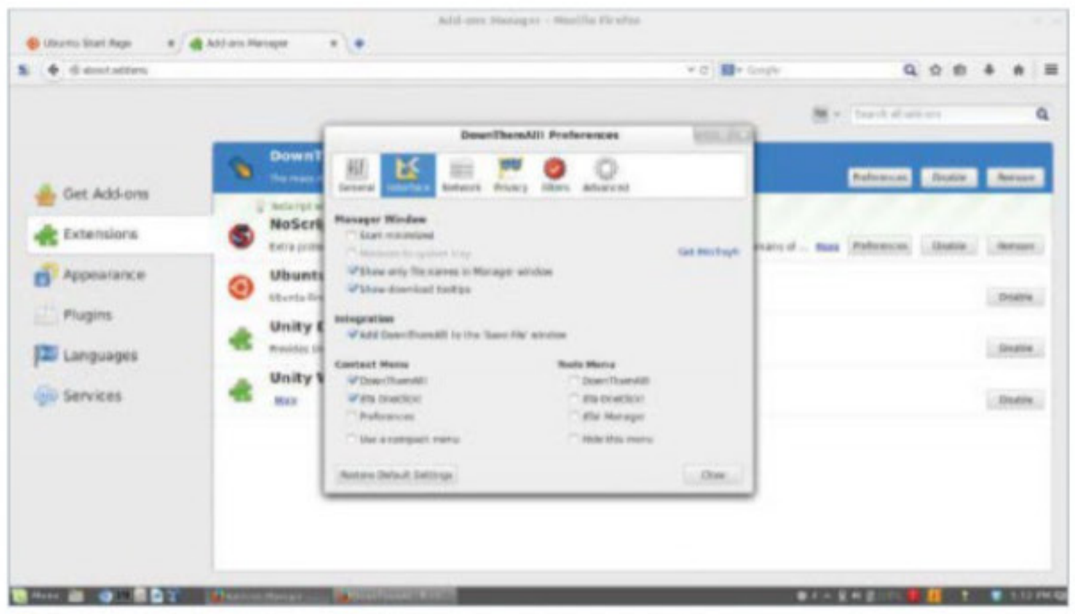
Overall

KGet doesn't really offer users a huge amount more than the download manager in the majority of popular browsers, although at least you can use it while the browsers are otherwise turned off.

6

DownThemAll!

Is the Firefox add-on good enough to make you switch browsers entirely?



There are actually a whole lot of options available for DownThemAll! that make it very flexible

Interface

Part of the integration in Firefox allows DownThemAll! to slot into the standard aesthetic of the browser, with right-clicking bringing up options alongside the normal downloading ones. The extra dialog menus are generally themed after Firefox as well, while the main download window is clean and based on its own design

Integration

It doesn't integrate system-wide but its ability to camouflage itself with Firefox makes it seem like an extra part of the original browser. It can also run alongside the normal downloader if you want, and can find specific link types on a webpage with little manual filtering, and no need for copy and pasting.

Features

With the ability to control how many downloads can happen at once, limit bandwidth when not idle and advanced auto or manual filtering, DownThemAll! is full of excellent features that aid mass downloading. The One Click function also allows it to very quickly start downloads to a pre-determined folder faster than normal download functions.

Availability

Firefox is available on just about every distro and other operating system around, which makes DownThemAll! just as prolific. Unfortunately this is a double-edged sword, as Firefox may not be your browser of choice. It also adds a little weight to the browser, which isn't the lightest to begin with.

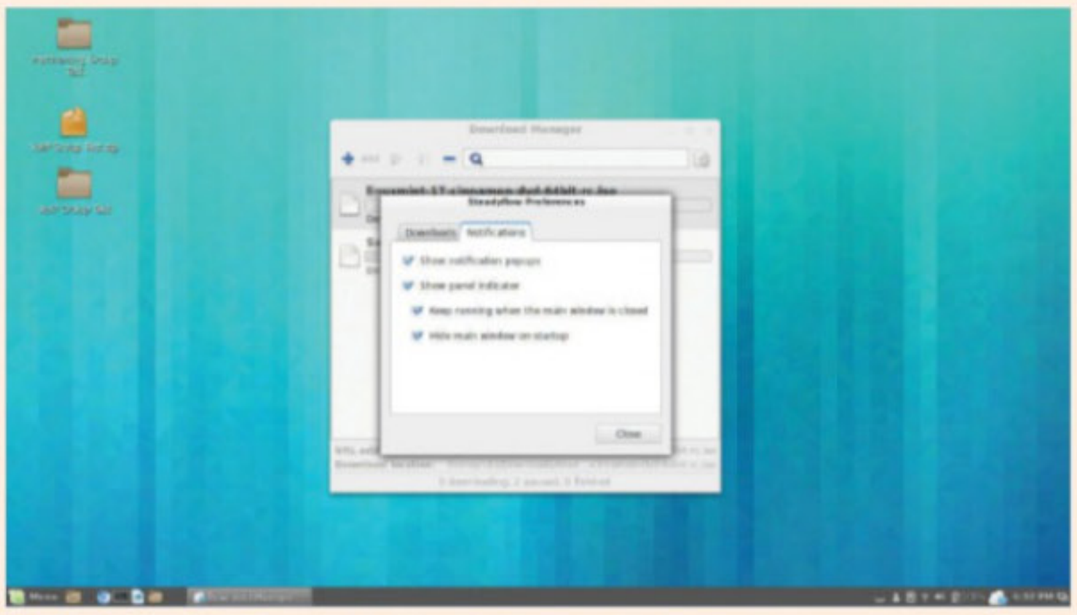
Overall

DownThemAll! is excellent and if you use Firefox you may not need to use anything else. Not everyone uses Firefox as their preferred browser though, and it needs to be left on for the manager to start running.

7

Steadyflow

Steadyflow has been gaining some traction – but for good reason?



The settings in Steadyflow are extremely limiting and somewhat difficult to access

Interface

Steadyflow is quite simple in appearance with a pleasant, clean interface that doesn't clutter the download window. The dialog for adding downloads is simple enough, with basic options for how to treat it and where the file should live. It's nothing we can really complain about, although it does remind us of the lack of features in the app.

Integration

Reading copied URLs is as standard and there's a plug-in for Chromium to integrate with that. Again, you can use FlashGot to link it up to Firefox if that's your preferred browser. You can't really edit what it parses from the clipboard though and there's no batch ability like in uGet and DownThemAll!

Features

Extremely lacking in features and the Options menu is very limited as well. The Pause and Resume function also doesn't seem to work – a basic part of any browser's file download features. Still, notifications and default action on finished files can be edited, along with an option to run a script once downloads are finished.

Availability

Only available on Ubuntu and there's no easy way to get the source code for the app either. This means while it's easily obtainable on all Ubuntu-based distros, it's limited to these types of distros. As it's not even the best download manager available on Linux, that shouldn't be too big of a concern.

Overall

Frankly, not that good. With very basic options and limited to only working on Ubuntu, Steadyflow doesn't do enough to differentiate itself from the standard downloading options you'll get on your web browser.

5

In brief: Compare and contrast our verdicts

	uGet		KGet		DownThemAll!		Steadyflow	
Interface	A clean yet informative interface with all the useful options available	8	A very simple, KDE-inspired aesthetic that makes the most of its features	7	Themed after Firefox, DownThemAll! is neither overly complicated nor too simple	8	Simplistic design showing the bare minimum functions needed	7
Integration	Needs some extra apps and extensions but reads the clipboard well	7	Only really integrates with Konqueror and partially with Firefox	4	Completely integrated with Firefox but nowhere else in the system	5	There are plenty of plug-ins and apps to get it working with browsers	7
Features	Filled to the brim with just about every function imaginable	9	A low feature set barely above the minimum we would expect	6	A large selection that allows for advanced batch download and one-click downloads	8	The smallest selection of features in this test, barely beyond browser capabilities	3
Availability	In multiple repos, available in several binaries and source	10	Not easily accessible outside of repos, and works best with KDE	5	Available wherever Firefox is, however you require Firefox for it	3	Only on Ubuntu and the source is difficult to come by	2
Overall	**An excellent download manager that truly aids your day-to-day browsing**	**8**	**KGet is fine but it really needs you to be completely in with KDE to work at its best**	**6**	**DownThemAll! is good but its connections with Firefox are a double-edged sword**	**7**	**Even if you have Ubuntu there are other much better apps available to use**	**5**

AND THE WINNER IS...

uGet

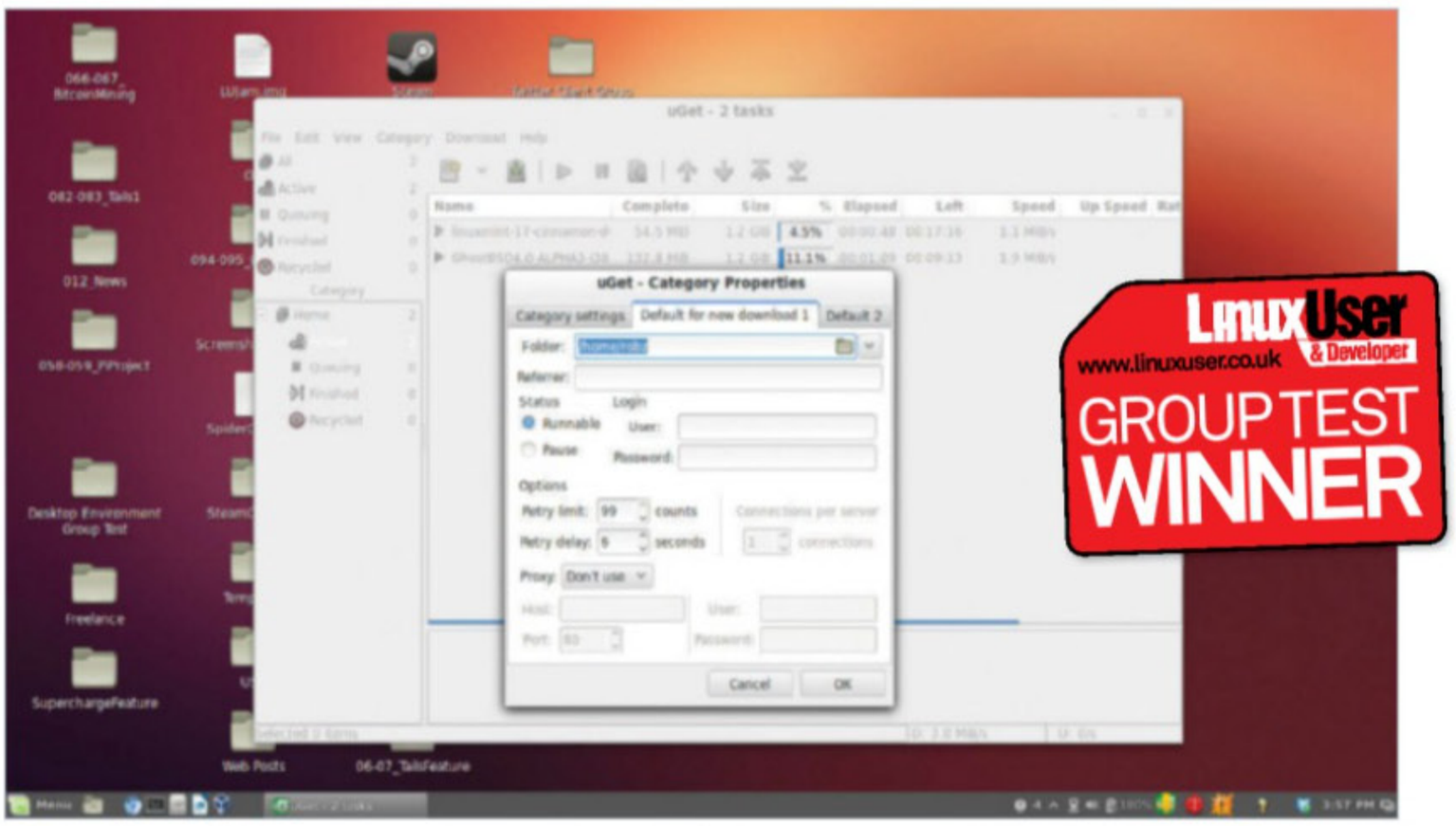
In this test we've proven that there is a place for download managers on modern computers, even if the better ones have cribbed from the torrent clients that seem to have usurped them. While torrenting may be a more effective way for some, with ISPs getting wiser to torrent traffic some people may get better results with a good download manager. Not only are transfer caps imposed by most major ISPs, some are even beginning to slow-down or even block torrent traffic in peak hours – even legal traffic such as distro ISOs and other free software are throttled.

Steadyflow seems to be a very popular solution for this, but our usage and tests showed an underdeveloped and weak product. The much older uGet was the star of the show, with an amazing selection of features that can aid in downloading single items or filtering through an entire webpage for relevant items to grab. The same goes for DownThemAll!, the excellent Firefox add-on that, while stuck with Firefox, has just about the same level of features, albeit with better integration.

If you're choosing between the two it really comes down to what your preferred browser is and whether you need to have downloads and uploads going around the clock. DownThemAll! requires Firefox running, whereas uGet runs on its own, saving a lot of resources and electricity in the process – obviously this makes uGet a much better prospect for 24-hour data transferring and it really isn't a major hassle to set up big batch downloads, or even just get the download information from your browser.

Give download managers another chance. You will not be disappointed with the results.

Rob Zwetsloot



Drill down different categories and download alternate types of files in different ways

UNLOCK THE POWER OF ANDROID

www.littlegreenrobot.co.uk

Android magazine

Available from all good newsagents and supermarkets

ON SALE NOW

› Hacker's Toolkit › Master Chromecast › HTC One Mini 2 review › Travel apps

HARDWARE REVIEWS

ANDROID HACKING

TIPS & TRICKS

APP REVIEWS

HELP & ADVICE

BUY YOUR ISSUE TODAY

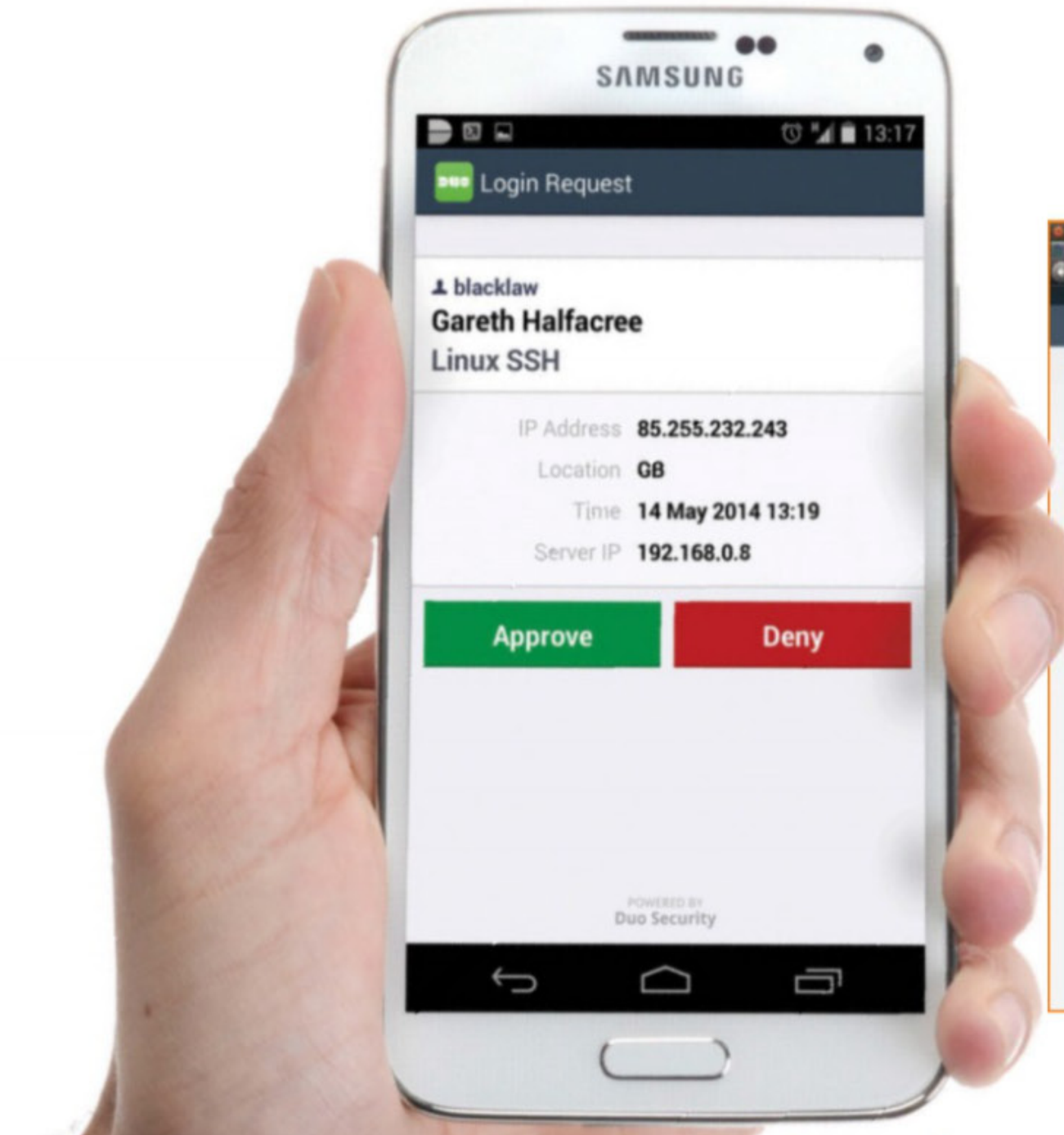
Print edition available at www.imagineshop.co.uk

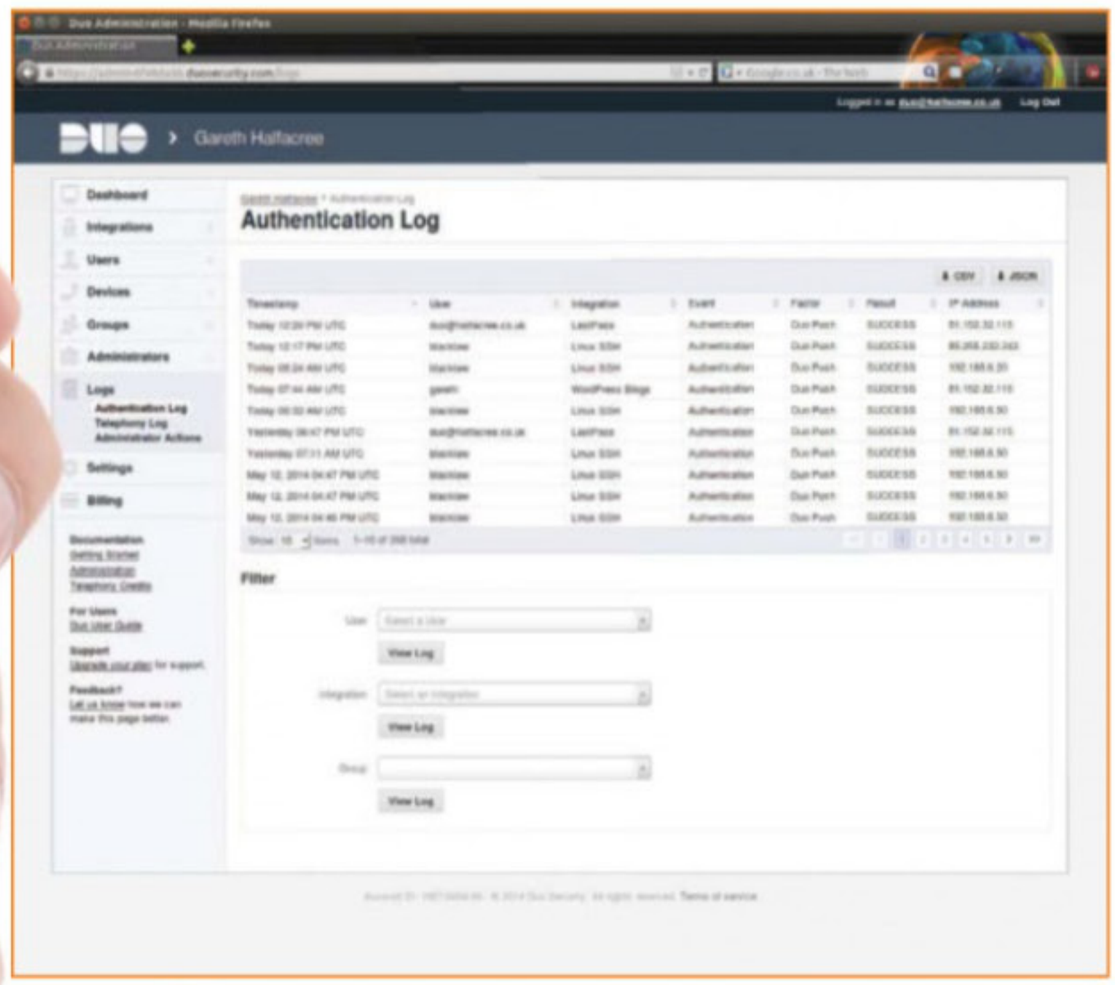
Digital edition available at www.greatdigitalmags.com

Available on the following platforms

facebook.com/littlegreenrobot

twitter.com/lgrobot





■ All authentication attempts, whether accepted or rejected, are stored in logs on Duo Security's server

APP

Duo Security

In the wake of Heartbleed, can a free smartphone-based two-factor authentication system offer peace of mind?

Pros
An affordable and powerful approach to two-factor authentication, requiring no additional hardware

Cons
Configuration can be daunting for non-enterprise users and the free mobile app has limited support

Traditional password authentication has long been recognised as the weak link in the security chain, even before the Heartbleed vulnerability exposed the private keys of millions of servers worldwide. The problem is that a password the user can easily remember is rarely a good password, while a good password is rarely easy to remember.

Two-factor authentication aims to fix this flaw. The most common implementations mix something you know with something you have: a password and a physical key, which uses a seeded pseudo-random number generator to verify that you are indeed the account holder.

It's neat, but awkward: the hardware tokens are typically expensive and various platforms require tokens from various vendors, leading to a full keyring. Duo Security is just one such company that is looking to solve that issue – and it appears to have come closer than most.

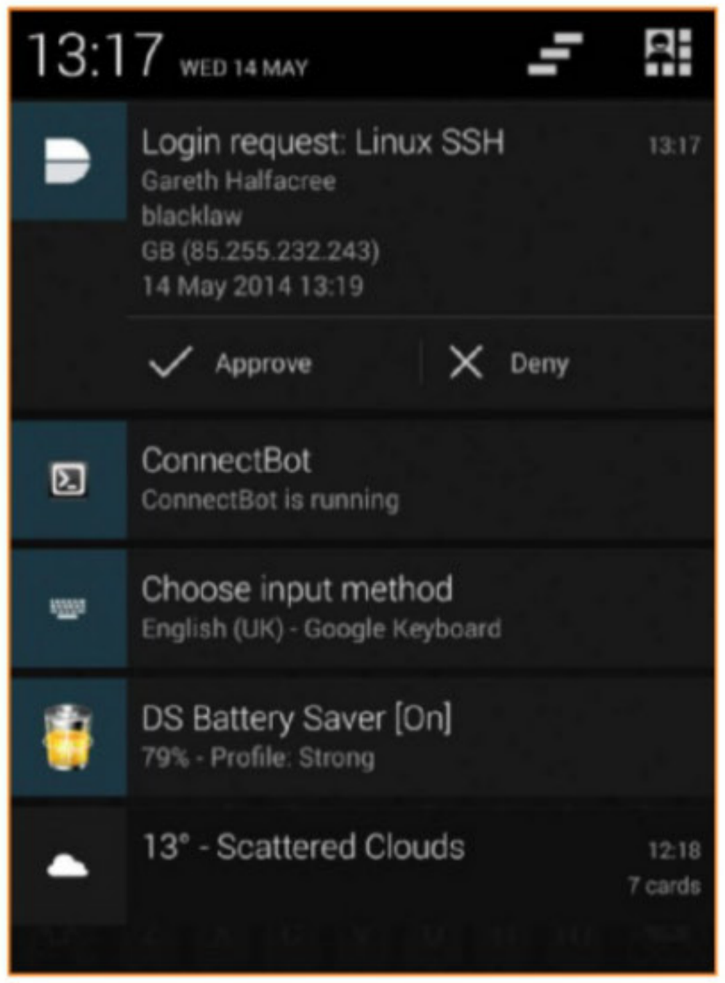
Signing up for a Duo Security account is free of charge if you're looking to protect fewer than ten users. For larger companies, the fee is $1 per user per month for an unlimited number of accounts, while the Enterprise Edition at $3 per user provides access to an application programming interface (API) for integration into bespoke applications.

The API is an especially impressive feature, allowing paying users to quickly add Duo support to their in-house applications. Those who just want to use the software with existing applications will find a long list of supported platforms: source code is provided for integration into Linux and other Unix-like systems, while modules are available to integrate with most brands of virtual private network (VPN) devices as well as a wide variety of software from the LastPass cloud-based password manager to the popular WordPress blogging platform.

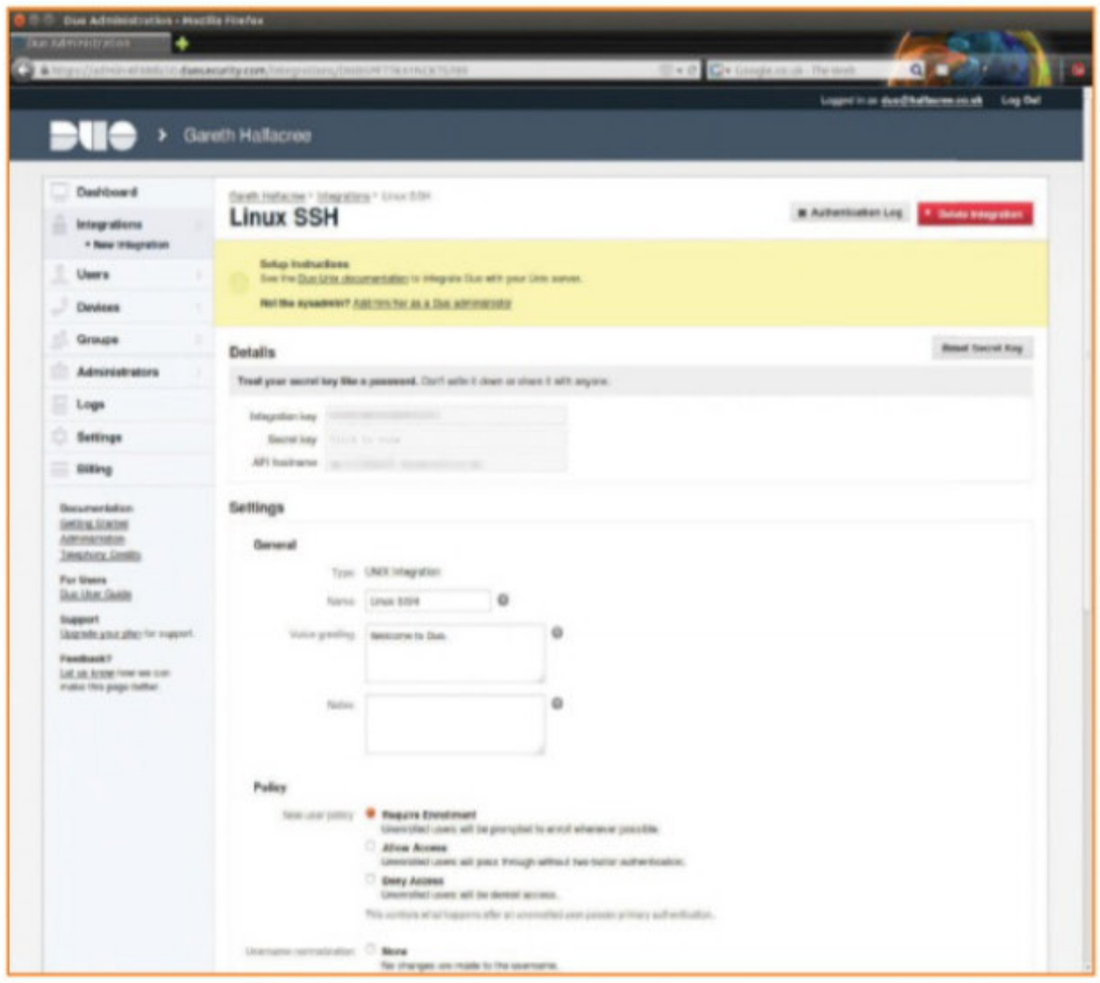
Duo Security

Can a free smartphone-based two-factor authentication system offer peace of mind?

REVIEW



The Duo Android app allows for logins to be authenticated from the notification panel



Designed primarily for enterprise users, the Duo web configuration can be daunting

“The Android app is particularly slick”

These modules can work independently, verifying the authenticity of each user's login attempts by sending a text message or making an automated voice call. Doing so uses up 'telephony credits', which can be bought in bulk or are automatically regenerated each month for paying accounts.

A better choice is to use the Duo Security smartphone application, available for Android, iOS and BlackBerry 10. This runs in the background and supports push notifications, which can be acted upon with just a simple tap.

The Android app is particularly slick: the phone is registered with the Duo Security management site by scanning a QR code, while authentication prompts can be confirmed or denied directly in the notification system without having to load the app itself.

For users who don't like the idea of tying their security into their smartphones, Duo supports traditional hardware tokens, available from the company for $20 each. A minimum order quantity of 20 units for international orders is somewhat annoying, although any third-party OATH HOTP token – including the popular YubiKey – can be used instead.

Duo's SSH server integration is simple yet powerful: compile the software from the provided source code and add it to the SSH config. As the software runs at log-on, users will be prompted to choose a means of authentication: app-based Duo Push, SMS or voice call. If SSH is being called non-interactively, such as by a backup script, Duo recognises this and defaults to the first available authentication method for confirmation.

If there could be anything bad to say about Duo Security, it's that the administrative website is somewhat user-unfriendly. While it's packed with features – including detailed logs of all authentication attempts across all services – it's clearly aimed primarily at enterprise users and may leave home users who are just looking for a bit of extra security feeling confused. It's well worth working your way through, though – and once your account is set up you'll rarely need to visit the site again.

Gareth Halfacree

Summary

It's hard to fault Duo Security. While points could be deducted for the software's lack of an official app for older or more esoteric phone platforms, or its somewhat user-unfriendly master configuration site, its power and flexibility more than make up for this. For home users, the free tier will be more than enough to bolster security.

More information

duosecurity.com

The front-facing status lights are basic, but provide at-a-glance monitoring of hard drive health and activity

The front of the chassis is fixed, with the drive bays only accessible from the rear of the DS414j

The metal lid of the DS414j includes the traditional Synology logo, serving as an air intake vent for cooling

The bulk of the DS414j's chassis is thick plastic, aside from the metal lid and rear fan mount

MINI PC

Synology DS414j £275

With a dual-core processor and four drive bays, has Synology got the perfect upgrade for those who have outgrown dual-bay NAS boxes?

Pros
Remarkably compact and very low power, the DS414j packs a surprising amount of functionality into its chassis

Cons
Its cut-down dual-core ARM processor and 512MB of RAM make the system unsuitable for the more resource-intensive applications on offer

When you buy a Synology product, you know what you're getting yourself into. The company's designs rarely change between generations – beyond a few small tweaks and improvements to the internals – and its Linux-based DiskStation Manager operating system only ever improves with time. Its pricing, however, can leave it out of the reach of the budget-conscious buyer, especially when more than two drive bays are required.

It's a problem of which Synology is only too aware and its response is the DS414j – the suffix indicating that the device is an entry-level NAS in its small and home office four-bay range. Designed for those who can't stretch to the DS414, corners have been cut: the Marvell Armada XP dual-core 1.33GHz processor and 1GB of DDR3 RAM have been replaced by a 1.2GHz MindSpeed Comcerto 2000 and 512MB of RAM – a combination that's more commonly associated with networking hardware than storage devices.

While the DS414j retains the four drive bays of its more expensive predecessor, further changes have been made: the chassis does not provide drive access from the front, with the sleds only accessible by undoing four thumb-screws at the back of the unit and folding down a flap containing the box's twin 80mm fans. That means no hot-swap support, and an awkward period of down-time if you need to replace a failed drive.

There are other areas in which the DS414j fails to live up to its higher-priced stablemate, too: the single gigabit Ethernet port and weaker processor means a drop in peak throughput from the DS414's 135MB/s write and over 200MB/s read to 80MB/s write and 112MB/s read.

It's easy to focus on the negatives, the areas that have been trimmed in order to bring the price down, but the DS414j is still an impressive piece of equipment. The change in processor means a considerable drop in idle power draw – from the DS414's 14W to under 9W – and a similar drop in active draw.

Synology DS414j

Has Synology got the perfect upgrade from dual-bay NAS boxes?

REVIEW

A pair of 80mm fans are whisper-quiet under normal use and mounted to a metal flap that provides access to the drive bays

A single gigabit Ethernet connection is an indication of the corners cut compared to the more expensive DS414 model

Two USB ports are provided, one USB 2.0 for peripherals and another USB 3.0 for external storage

Technical specs

Operating System	DiskStation Manager 5.0 (Proprietary Linux-based)
Processor	MindSpeed Comcerto 2000, Dual-Core 1.2GHz ARM Cortex-A9
Memory	512MB DDR3
Dimensions	184mm x 168mm x 230mm
Weight	2.21kg
Drive bays	4x 3.5-inch SATA 3.0Gb/s with sleds, 5TB maximum capacity per sled
Networking	1x Wired Gigabit Ethernet
Expansion	1x USB 2.0, 1x USB 3.0
Cooling	2x 80mm fans, 18.9dBA

Also consider

HP ProLiant MicroServer G7 N54L
£179.90
Frequently available with cashback offers that drop the cost below £100, the AMD Turion-based HP MicroServer offers plenty of capabilities. Its processor is far more powerful than the Comcerto, its memory is upgradeable and it can be flashed for hot-swap support – but it is considerably larger than the DS414j.
serversplus.com

Asustor AS-604T
£475.02
Although far from cheap, the Asustor AS-604T offers a powerful 2.13GHz Intel Atom processor, front-accessible hot-swap drive bays, a handy status LCD and even the ability to stream media via an included HDMI connection. Memory is also user-upgradeable via SODIMM, to a maximum of 3GB from the bundled 1GB.
amazon.co.uk

“The DSM 5.0 software is a pleasure to use, with an attractive GUI that runs entirely in a browser”

The DS414j is considerably cheaper than the £352 DS414, and while it comes with a few licensing and performance restrictions – 1,024 user accounts to 2,048, two folder sync tasks to four, 128 maximum concurrent CIFS connections to 256 and so forth – it's more than capable of the majority of tasks required of a small-office NAS.

Performance aside, the feel of the DS414j is exemplary: the DSM 5.0 software, the latest release of Synology's custom Linux distribution, remains a pleasure to use with an attractive graphical user interface that runs entirely in a browser. Built-in software allows the DS414j to talk almost any common network protocol and, as usual, there are plenty of add-on packages available to install, including the popular WordPress blogging platform.

It's here, sadly, that Synology demonstrates that it hasn't learnt from its previous releases: installing packages through the easy installation wizard results in an out-of-date version of the software – with WordPress and other public-facing packages often the target of hackers, it can leave the NAS and its files worryingly underprotected.

It's hard not to like DSM, though. It's one of the best NAS operating systems around and something that really makes Synology's products stand out from the crowd. It's also possible to manually install packages, so you can ensure that you have the very latest versions on offer.

The DS414j targets a tricky market: those with decent budgets will find the DS414 a far more suitable choice, while those counting the pennies can get much of the same functionality from devices costing as little as £100 – albeit without Synology's polished DSM software.

Gareth Halfacree

Summary

The newest DiskStation Manager software continues to impress, but its one-click installation of add-on packages can leave you outdated and unprotected. The performance of the ARM-based processor also means some apps will perform sluggishly. However, there's no denying the DS414j excels at its core network file storage tasks.

More information

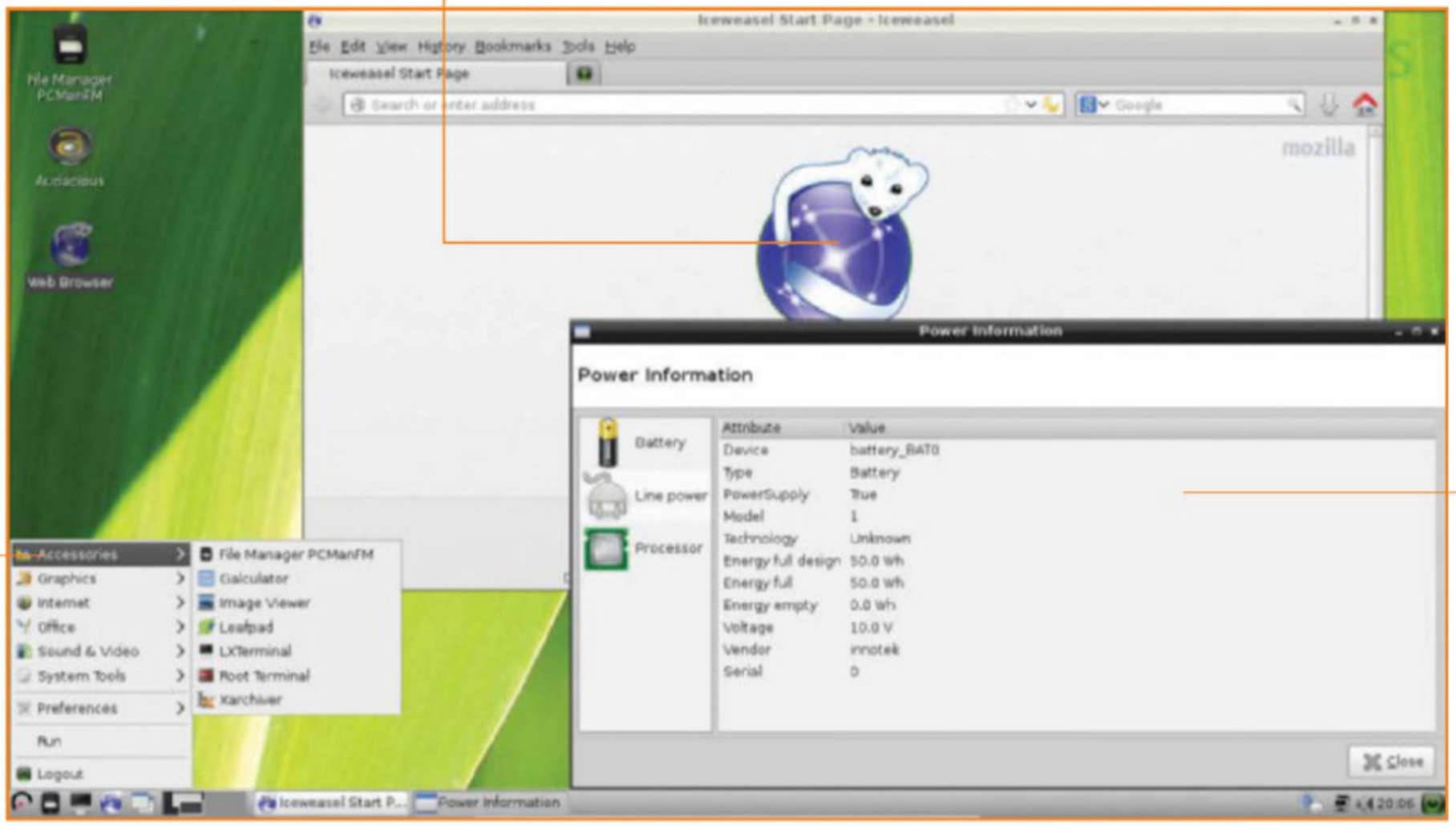
www.synology.com

New features: Debian base • Better wireless drivers • New installer • Iceweasel browser

The selection of apps is extremely low; there's not even an office suite to start with

On the surface, not much has changed since the switch to Debian, however Iceweasel is the most noticeable

wattOS cares about your battery and power usage, and is one of the most energy-efficient distros around



DISTRO

Best for: Laptops

wattOS R8

Minimum Specs: CPU Pentium III RAM 128MB STORAGE 3GB

The green distro makes a switch from Ubuntu to Debian. Does it make the lightweight distro better or will we return to wattOS 7.5?

Pros
Lightweight and energy efficient, wattOS is also very easy to use, with few sacrifices made to achieve this

Cons
Installation leaves one of the trickiest parts as a manual operation and the image size is still quite large

Lightweight Linux distributions are inherently energy saving. By definition, you're using fewer resources to run your system, which in turn requires less power and electrical draw. Throw in some power-efficient hardware and idle power draw will be minimal. These lightweight systems – while naturally energy-conserving – don't normally include any specific optimisations for power saving. This is where wattOS comes in.

While also lightweight, wattOS strives to strike a balance between conservative code and usability. The net result is a little less wattage while idle and a longer-lasting laptop battery when disconnected. It's the usability part that is very important to wattOS: something like Puppy Linux or Tiny Core may likely be less resource-intensive while idle, however you need to make some level of sacrifice regarding the desktop and available software to use these distros.

To further its goal, the newest version of wattOS has switched to the current stable branch of Debian 7.0, Wheezy. Before version eight the distro was running on Ubuntu, stripping away many of the core components. With Debian the team can actually build it up more than strip it down, making for a better product overall.

Debian differences

wattOS still comes in three main flavours: LXDE, MATE and the ultra-slim Microwatt spin that has switched from PekWM to Openbox. Openbox is about as light as PekWM but a little more popular and better supported, resulting in a better overall experience while still providing maximum power benefits.

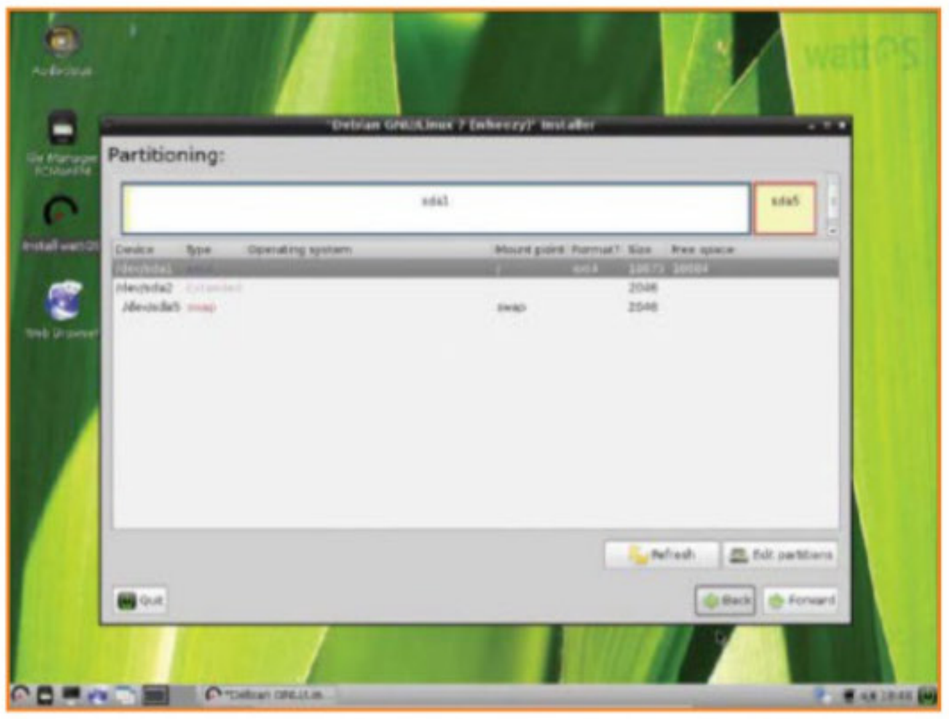
The LXDE version remains the flagship version of wattOS, sitting in between MATE and Microwatt as a perfect balance

wattOS R8

Has the switch from Ubuntu to Debian improved this lightweight distro?

REVIEW

Although the default install comes with a low app count, it can be easily inflated with access to Debian's repos



Installation is generally fine but the manual partitioning seems like the method of yesteryear

"We found no decrease in battery life over our course of testing the distro"

in terms of required resources and usability. The switch to Debian hasn't reduced the size of the images, unfortunately – this was a concern we had with the last round of wattOS releases and the ISOs for R8 are even larger than the ones found in R7.5. This is a minor issue, but one that can be important for lightweight distros.

Installing from these images is not as easy as some of the more major distros. Setting up your user account, time zone and other little things are kept simple, yet you are required to manually partition your hard drive. This is no big deal for a lot of Linux users, but there's no real description or instructions on how the hard drive should be laid out. This can very easily confuse newer users or those used to the ease of Fedora, Ubuntu and other modern browsers; it's an unnecessary barrier to entry in a landscape where excellent and easy-to-use installers are the norm.

Lightweight software

The main difference you'll notice software-wise with the switch is the use of Iceweasel over Firefox; this is the standard Debian alternative to the quintessential open source browser. The main difference is that, while based on Firefox, it doesn't receive the same level of constant updates and retains an older aesthetic to the overall design. While Midori may be a more traditional choice for a lightweight distro, Iceweasel is much less resource-intensive than the full version of Firefox.

It's flanked by a small selection of other light apps such as Audacious for music, VLC for video and a basic PDF reader. The entire system takes up less than three gigabytes all together, however you have full access to the rest of the standard Debian packages via the Synaptic Package Manager.

All of this allows the distro to boot very fast, even on older hardware. Within seconds we were at the login screen on a more modern system; loading apps and general browsing was fast and smooth, and the memory footprint stayed fairly low relative to other distros. Most importantly, we found no decrease in battery life over our course of testing the distro. Without more thorough testing we'd suggest it was a touch better than with Ubuntu, however it will entirely depend on your workload as well as the load on the system.

Green fingers

The latest version of wattOS has managed to keep to the high standards of the previous few versions, even with this major shift to Debian. It's not without its gripes, though: the installer should be better and the ISO could probably go on a diet to lose an extra few megabytes.

These issues aside, the presentation, speed and day-to-day product of wattOS is a solid and lightweight distro that truly cares about power usage and the user. If you're looking for a new distro to power an older laptop – or even any laptop in general – this is an excellent option.

Rob Zwetsloot

Summary

Some minor issues aside, wattOS does what it sets out to do perfectly. It's stable, fast and generally very easy to use, which is in no small way thanks to the new Debian base. Get it for your laptop now.

Download now

www.planetwatt.com

Subscribe

Subscribe today & start saving
Non-USA readers see page 32

USA special offer

Subscribe today and get 5 free issues*

Exclusive subscriber benefits

- Subscribe today and pay just $130 for 13 issues*
- Pay as little as $10 an issue – usually $16.99 in stores

For more information and to order visit
www.imaginesubs.co.uk/lud
Or call **+44 (0) 1795 418661**

Imagine Publishing publishes more than **20 monthly magazines**, some of which have been running for over **10 years**, reaching over **4 million readers** every year

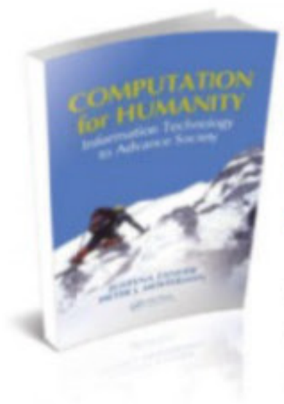
***Terms and conditions:** This is a US subscription offer, please don't forget to quote USA when ordering. You will actually be charged £80 sterling for an annual subscription. This is equivalent to $130 at the time of writing, although the exchange rate may vary. Five free issues refers to the newsstand price of $16.99 for 13 issues being $220.87, compared with $130 for a subscription. Your subscription will start from the next available issue. This offer expires 31st October 2014.

Learn To Program With Scratch

Author	Majed Marji
Publisher	No Starch
ISBN	978-1593275433
Price	£23.99
Score	4

Scratch's rich learning environment has proven to be a popular way into programming for young learners of all ages. Here Marji really pushes the language's capabilities in this project-based intro to real programming through the friendliest – and most fun – of environments.

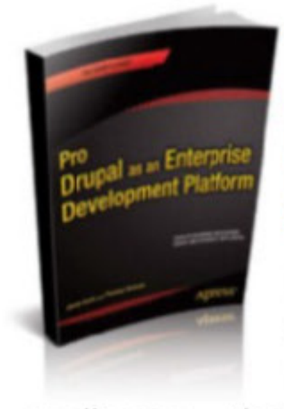
Computation For Humanity: Information Technology To Advance Society



Author	Justyna Zander & Pieter Mosterman
Publisher	CRC Press
ISBN	978-1439883273
Price	£89.00
Score	4

A mixed bag, but full of interest. Skip the (academic) introductory first section and encounter chapters on computing for sustainability (and some fascinating facts); Engineering and Discovery (Robots; the Quantum Frontier; Biologically-Inspired Real-Time Self-Organising Systems); Citizen and Artisan Computing; and Visionary Pondering! One for your local hackerspace's library.

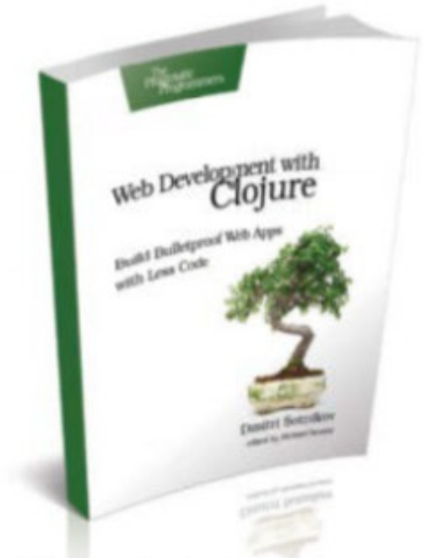
Pro Drupal As An Enterprise Development Platform



Author	Jamie Kurtz, Thomas Besluau
Publisher	Apress
ISBN	978-1430260042
Price	£35.50
Score	4

It seems to be a sign of changing times – DevOps, Agile, cost-of-quality gap – that a PHP CMS is proposed as an enterprise platform. Kurtz and Besluau make a good case, with 20,000 modules and a stable, secure Drupal core making a strong base for apps to integrate into the larger enterprise ecosystem.

Web Development With Clojure: Build Bulletproof Web Apps With Less Code



Author	Dmitri Sotnikov
Publisher	Pragmatic Bookshelf
ISBN	978-1937785642
Price	£27.50
Score	4

A powerful, functional web development intro

Clojure works the functional way, so it's no surprise that for web development it doesn't follow the kitchen sink web framework approach. You get the powerful Ring and Compojure libraries, which avoid most boilerplates and leave you to write clean, maintainable code.

Using a guest book and picture gallery example, Sotnikov shows you how to build upon the Ring HTTP library with the Compojure library, in order to provide the handler functions and create multiple routes. JDBC is used to connect to relational databases but an alternative is also shown, using a Clojure Domain Specific Language (DSL). Throughout this book, alternative components are offered, driving home the message that Clojure is an extremely flexible choice for your projects.

The practical side of development life – unit tests, packaging, logging – has not been neglected; using a REPL means instant results. The useful appendices include an examination of IDE choices; a fine short introduction to the Clojure language; plus CouchDB and MongoDB.

Advanced Topics In C: Core Concepts In Data Structures

Author	Noel Kalicharan
Publisher	APress
ISBN	978-1430264002
Price	£30.86
Score	4

Absorbing programming tutorials

From the fertile mind of a remarkable teacher, Dr Noel Kalicharan, comes a guide to programming concepts *using* C, rather than just advanced concepts *in* C. By using the former it goes a long way to accomplishing the latter, preparing the reader for the growing demand for C coders in embedded and IoT applications.

This work is written to C99 standards, using Code::Blocks, but is fine with GNU's suite of C tools. The chapters cover Sorting; Searching and Merging; Structures; Pointers; Linked Lists; Stacks and Queues; Recursion; Random Numbers, Games and Simulation; Working with Files; Binary Trees; Advanced Sorting; and Hashing, so it dives into every topic needed for the beginner-to-intermediate C programmer to get a broader understanding of programming.

Exercises without answers are a deliberate way of pushing the reader to write the program and therefore think through the problems. Kalicharan has an easy writing style that draws the reader in and it's great to see topics like recursion given decent coverage in a C book.

Speaking JavaScript: An In-Depth Guide For Programmers

Author	Dr Axel Rauschmayer
Publisher	O'Reilly
ISBN	978-1449365035
Price	£30.99
Score	4

An invaluable JS guide and reference

As Dr Rauschmayer overcame initial scepticism and fell in love with JavaScript, he chronicled his discoveries. His observations will take you deep into the language, and there's much here of interest even to experienced JS coders.

The Quick Start section gives you enough JavaScript to get productive, enabling coders proficient in other languages to join the party. The rest of the book first covers the language's background – the what, how and why, with all its quirks (there are many quirks, admittedly) and subtleties. For example, why does JavaScript have both undefined and null?; understanding the arbitrary distinction between primitive values and objects; overcoming 'too flexible' arrays; why JS' lack of integers is largely ignored by JS engines; and all the wonderful features that JavaScript has 'borrowed' from other languages.

The central In Depth section is the one you'll use as a reference for years to come, while the closing chapters are filled with all the advanced techniques, best practices and learning resources you need. A highly recommended read.

Contact us...

Web: www.linuxuser.co.uk
Email: linuxuser@imagine-publishing.co.uk

Facebook: Linux User & Developer

Twitter: @linuxusermag

Your questions answered

Send us your questions and we'll do our best to answer them!

FAQ: Purchasing a slice of Pi

Q: Where can I buy a Raspberry Pi?
A: There are a number of places the Pi is sold. Officially the main suppliers are RS Components and element14, but you can also get them from accessory sites such as ModMyPi or in kits such as the Kano.

Q: Where can I buy a camera board?
A: The Raspberry Pi camera module can be a little bit more difficult to track down but, again, it's mainly sold via the official Raspberry Pi outlets, such as RS Components and element14.

Q: Where can I buy other Raspberry Pi project components?
A: Pi community websites like Pimoroni and ModMyPi are your best bet for excellent parts. You can also get physical modules from Adafruit and everything else from Ciseco and the aforementioned element14 and RS Components.

■ Raspberry Pi components are readily available on a number of sites

Synaptic issues

I'm wondering if you guys could help me. I did a fresh install of Lubuntu 13.04 and opened Synaptic to check out which versions of some libraries were included, but I was quite surprised to see that only the installed packages were showing up. I tried to use **apt-get update**, but it showed up a lot of errors in the process. Mostly it seemed to be unable to download information from the packages and it showed up a line like this:

```
W: Failed to fetch http://pk.archive.
ubuntu.com/ubuntu/dists/raring-
backports/multiverse/binary-i386/Packages
404  Not Found
```

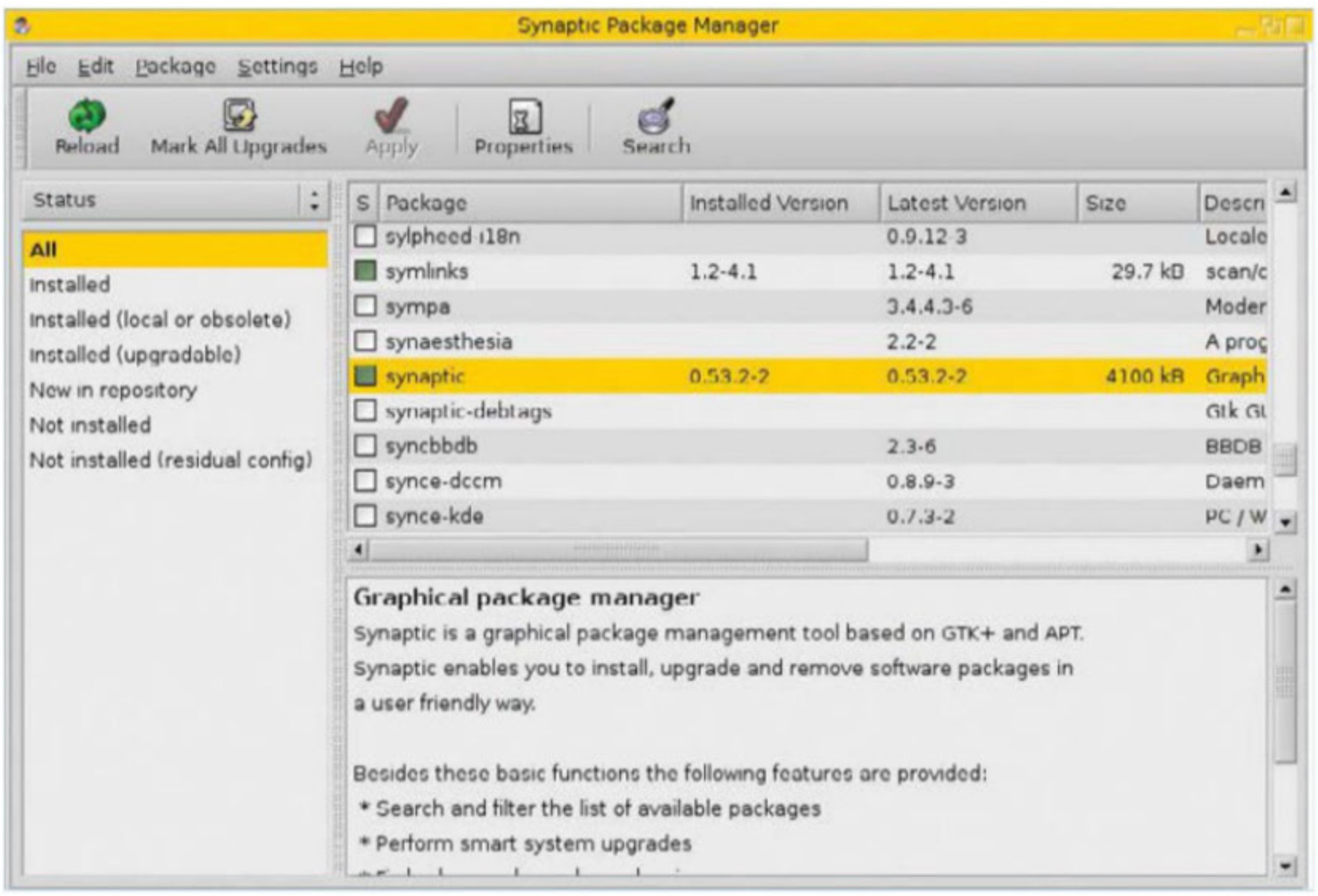
And then ended with this:

```
E: Some index files failed to download.
They have been ignored, or old ones
used instead.
```

I'm really unsure of what's going on here – is there anything you suggest to fix it?
Chris Harrington

This could be a couple of things, but clearly it's unable to contact the mirror for the packages. First of all, and it may sound silly, but make sure your internet is working fine and there are no problems connecting with your system. See if you can ping those mirrors from another system that you know works.

If that isn't fixing the problem then you may need to change which mirror you use. The



■ Synaptic won't work if the package lists can't be found

*quickest way to do this is to remove the pk from all the mirror URLs. You can do this by opening the **/etc/apt/sources.list** from the terminal in a text editor and just editing each line manually.*

If you're still having issues, try and reinstall and see if that fixes the issue.

Double the audio

Hi there. For some strange reason all the music files in one of my directories repeat themselves twice – as in the files are twice as long and contain each track twice in them. Perhaps I unwittingly merged them when restoring a backup or something.

In any case I have a question: is there a way to run FFmpeg or some other tool to cut my double MP3 files in half? The solution needs to be dynamic – that is, the program needs to measure the individual length of each MP3 file and find out by itself where the halfway mark is.

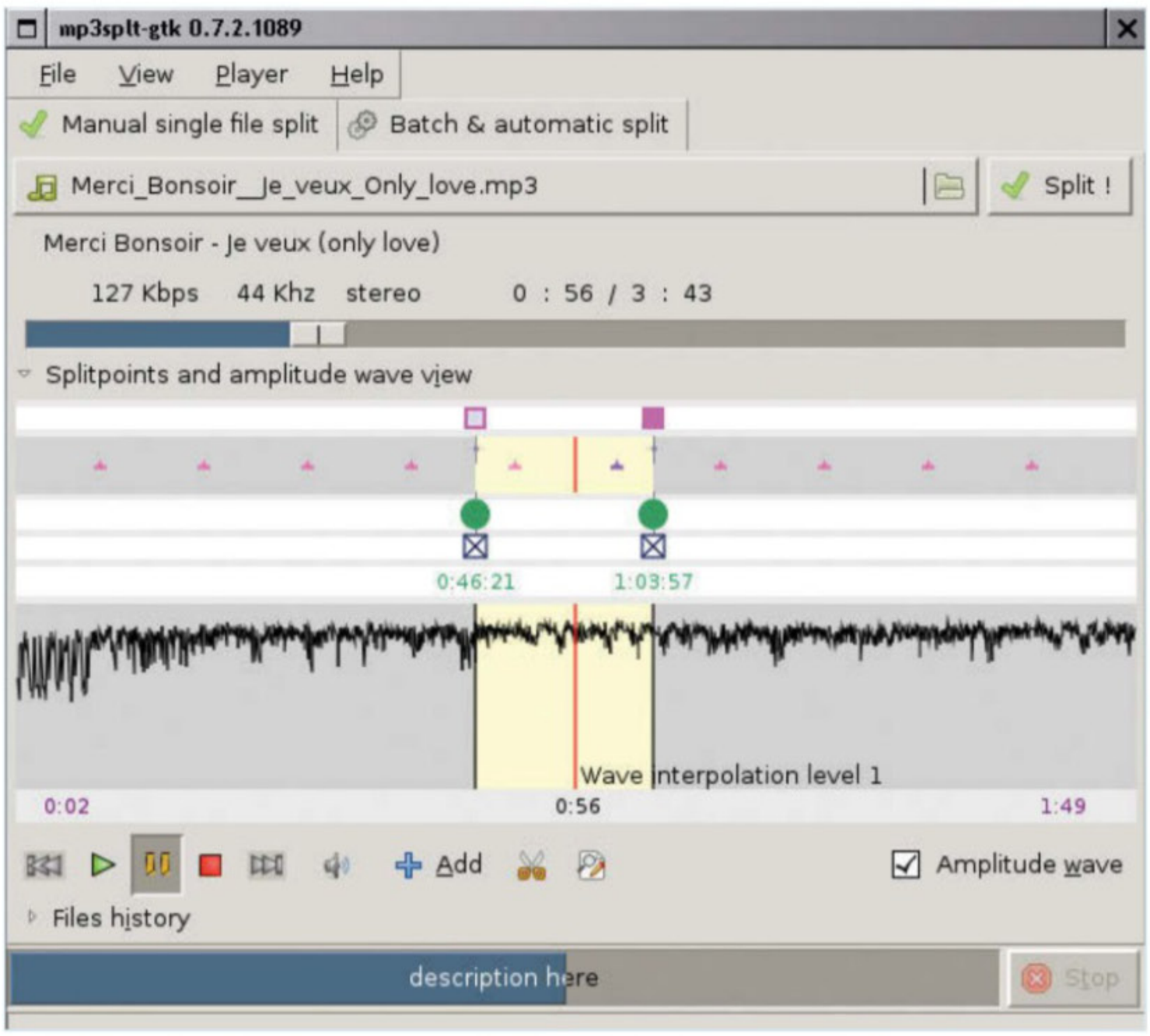
I could always do this manually in Audacity, but there are hundreds of sound files affected, so this would take me forever. Any help on this would be greatly appreciated.

Samuel Bean

That's an odd problem you seem to have there but you're right, there is a way to do it in FFmpeg that will do a batch process for any MP3 in a directory – although we need to create a script ourselves to handle it:

```
#!/bin/bash
mkdir unchopped
for file in *.mp3
do
mv -v $file unchopped
ffmpeg -i unchopped/"$file" -t $(echo
$(mp3info -p %S "$file")/2 | bc -l | cut
-f1 -d.) $file
done
```

This won't do any analysis on the file though, so if the halfway mark isn't what you want to split from, you might be in trouble; if you have to go to the effort of finding out where the split is for individual files you might as well use Audacity. For a slightly faster method of loading MP3 files so you can do it graphically is using MP3Splt.



■ The MP3Splt tool does not decode audio files, making the process a bit faster

Password behaviour

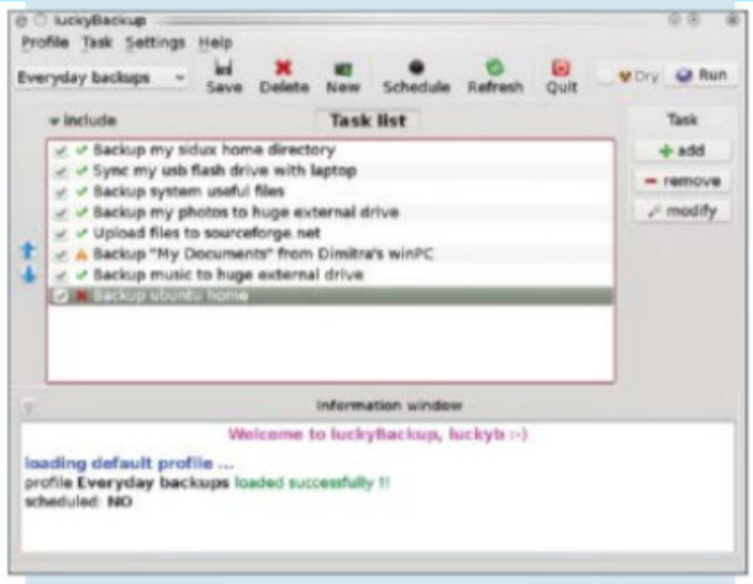
Hey, I'm trying to find a way to copy files to a couple of my servers throughout the day using a script but I'm baffled as to how to do it. The part that I can't figure out is how to pass the password to the remote server. I'm using a generic **scp** command:

```
scp *.* jobee@192.168.1.7:/mnt/bigdrive/
Backups
```

How do I include the password so the script will work without user intervention? Googling this is no help. Well, there was one hit that looked like it might do what I wanted it to do, but after I got to the third page I decided to post a question here. Is this possible? If so, how?

Thanks,

Joe Barnes



■ luckyBackup works over the network and might be slightly easier to use

*Right, what you need to use is **sshpass** for it to send the password over SSH. You will need to modify your command by using something like the following:*

```
sshpass -p 'password' scp -r [Path
to folder to backup] root@192.168.1.7:/
mnt/bigdrive/Backups/Docs
```

Obviously it's a little insecure to have the password for the root account just in the script or bash, so you should probably think of making an alternate account or encrypting it in some way. Hope this helps!

Questions & answers

Facebook: Linux User & Developer

Twitter: @linuxusermag

Partition woes

I'm currently in the process of installing Arch using the 2013.05.01 image on an UEFI system. Because cfdisk doesn't support GPT disks, I found out that instead I needed to use GNU Parted – which is version3.1 in my setup. However, I have not been able to successfully partition my hard drive.

Basically, I'm trying to create the swap with this in the command line:

```
mkpart primary linux-swap 2048s 4GB
```

This runs with no complaints from GNU Parted. When I print out the partition list though, it's not shown as a swap, but as a FAT16 file system. No matter which file system I try and create the partition as, it does the same thing – and any file system I create afterwards has no file system. I'm at a loss as to what to do. Do I even need to be doing this myself at this point of the install?

Gary Napier

■ Always make sure you partition properly

*With Arch, we would suggest maybe using **mkfs** and **mkswap** instead. Parted will create the partitions easily enough, but there seem to be some issues on some systems where it won't properly set them up. Otherwise, if the swap is the final one for you to make, then you can probably just use **mkswap** on the partition to convert it. The UEFI system will require a FAT32 boot partition, which you can create with something like:*

```
$ mkfs.msdos -F 32 /dev/sda1
```

Hope this helps!

The RAID

I'm currently running an RHEL 6.2 install with LSI hardware RAID controller. Approximately every ten minutes, an error message shows up in my syslog – while it's not really causing any major issues, it sometimes floods the logs and makes searching inconvenient when I need to. The message is:

```
hostname mpt-statusd: detected non-
optimal RAID status
```

When checking manually, **mpt-status** reports everything as okay:

```
[root@hostname ~]# mpt-status -i 5
ioc0 vol_id 5 type IM, 2 phy, 118 GB, state
OPTIMAL, flags ENABLED
ioc0 phy 1 scsi_id 6 ATA      MTFDDAK128MAR-
1J MA44, 119 GB, state ONLINE, flags NONE
ioc0 phy 0 scsi_id 7 ATA      MTFDDAK128MAR-
1J MA44, 119 GB, state ONLINE, flags NONE
```

I have searched online and have only found people without hardware RAID controllers experiencing the issue and they stop/disable **mpt-status** to resolve it.

This is not an option for me as I need **mpt-status** to be able to monitor my RAID array in case of any failures on the disks, etc. I have tested it and it plugs in perfectly with my monitoring system, and I would really like to be able to continue using it.

If you have any suggestions on how I can stop these messages from flooding my syslog for no good reason, it would be greatly appreciated.

Stephen

Right, even though this is a fairly simple message, it takes a bit to fix. Although if you're banging your head against a wall over it, it will probably be worth your while.

*First, you need to identify which SCSI ID **mpt-status** needs to check the RAID array on with the **-p** switch with:*

```
# mpt-status -p
Checking for SCSI ID:0
Checking for SCSI ID:1
Checking for SCSI ID:2
Checking for SCSI ID:3
Found SCSI id=3, use ''mpt-status -i 3``
to get more information.
```

It will list a few IDs and then give you a command to get more information, which will look something like:

```
# mpt-status -i [ID]
```

Once you have the ID information, edit /etc/init.d/mpt-statusd. Change the0 the "ID=0" to the ID that you got from the steps above:

```
MAILTO=root    # Where to report
 problems
  PERIOD=600     # Seconds between
   each check     (default 10
   minutes)
```

```
REMIND=7200    # Seconds between each
reminder (default 2 hours)
RUN_DAEMON=yes
ID=[ID]
```

You can also create a config file in location /etc/default/mpt-statusd with the configurations options above. This would make your config 'immune' to being overwritten if you updated the package at any time.

Upstart and Mint

Normally I dislike GUIs for simple tasks. But I find that my Linux Mint system uses this Upstart startup thing that I don't really care for. Is there some quick and dirty way to see what is tasked to run when, and disable/enable things easily?

I am so used to dealing with **/etc/init.d** and **/etc/rcX.d**. I can easily see what is being started at any given runlevel by doing an **ls** command on the appropriate directory and looking for files that start with a capital 'S'.

Now I find that most of my system services use Upstart. I really don't have any desire to learn the syntax of the config files (although it appears simple), or learn a new way to figure out what is or is not tasked to run at different runlevels. I think **ls /etc/rcX.d** was perfectly fine for that need.

Example: I find that smbd is running on my system. I don't want that. And I can no longer do the simple thing:

```
mv /etc/rc2.d/S50smbd
/etc/rc2.d/s50smbd;
/etc/init.d/smbd stop
```

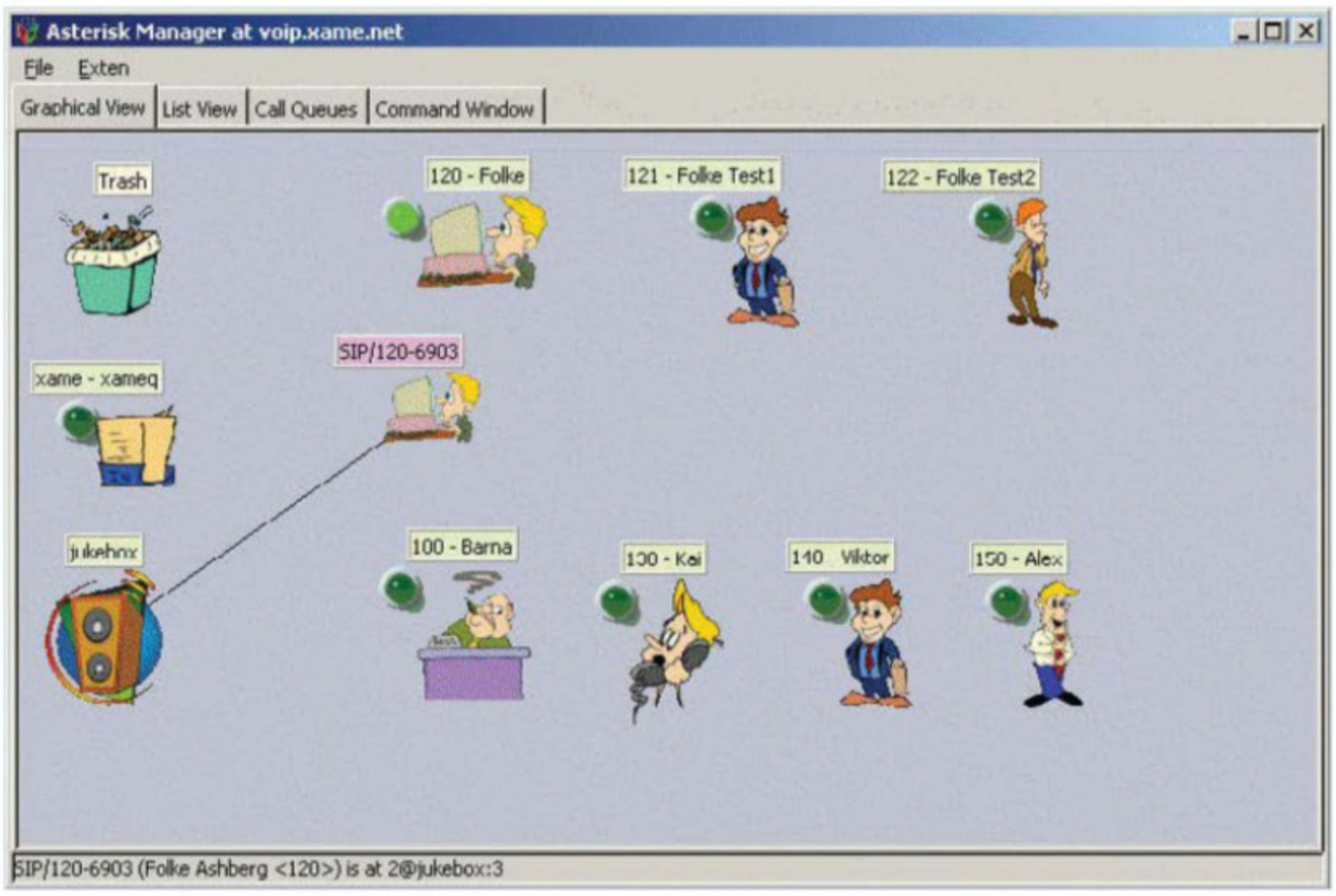
Does Upstart now mandate that I go into the **/etc/init/smdb.conf** file and hack away at its internals to stop smbd from starting?

I should probably go and learn Upstart anyway. It's the same thing as me preferring old GRUB to new GRUB though. Why change if you see no benefit and don't want to? Upstart appears to be what Ubuntu and its derivatives like to use – I see no need for the change.

Tommy

*Well you won't need a graphical interface, as you can fix it by creating a file with the same name, but appending it with **.override** in the **/etc/init/** folder. You can do this with something like this:*

 # echo manual > /etc/init/[service].override; chmod 644 /etc/init/[service].override



■ Asterisk can be a useful VoIP service when working correctly

Upstart was basically created to deal with a few things that are different on modern computers compared to how they used to be – events are dealt with asynchronously rather than queuing it up. Inserting and removing storage is much more common with USB storage these days, so it handles this well without having to lock up the system. It's in theory a convenience, but obviously it does change things you may already be used to.

Remote Asterisk

I've got an Asterisk voice calls archive located at **/var/spool/asterisk/monitor** on an ext3 filesystem. I'm trying to use SFTP provided by SSHD from a C# app on an external machine to move files from this above directory to another disk, which is also ext3 and mounted at **/mnt/voicearchive**.

I have got 777 permissions set up on both the relevant directories. The problem is that the SFTP user apparently cannot move files – I literally pass the following...

```
mv /var/spool/asterisk/recording.gsm /
mnt/voicearchive/recording.gsm
```

... to SFTP / SSHD and I get back 'Failure'.

So, I tried FileZilla, logging into SFTP and dragging the file from **/var/spool/asterisk/recording.gsm** to **/mnt/voicearchive** – it also passes the **mv** command and returns as 'Failure'.

However, if I open an SSH session in a terminal and execute the **mv** command as is, just as I pass it from C# via SSH or via FileZilla, the file moves, no problem.

Any ideas why I cannot move the file using SFTP? I'm running on a CentOS 6 box, both disks that contain each filesystem are mounted as ext3 and I've got 777 rights in both locations (first thing I thought was permissions).

Henry Ardwyn

SFTP doesn't support moving files across mount points that are on different devices. It uses the rename action to basically move files, changing its location to the target in the process.

*We suggest using **symlink** in the 'local' folder, and use that to perform actions across the mount points. So, if you have something like **/var/spool/asterisk/recording**, which resides in **/dev/sda1** and voicearchive in **/dev/sdb1**, you can do something like this:*

```
$ ln -s /mnt/voicearchive archive
```

*...while in **/var/spool/asterisk/recording**.*

From its perspective, it will then be able to easily rename/move the files into the new mount folder. However, if it's mounted locally as such, you don't specifically need to use an FTP protocol to get it done.

Hosting listings

Sponsorship opportunity
Bring attention to your brand by sponsoring this section. Contact Rhian Carter on +44(0)1202 586421

Got a deal you think we should list?

Whether you're a hosting firm or a happy customer who wants a favourite provider listed, drop us a line with the details!
linuxuser@imagine-publishing.co.uk

Dedicated server listings

NAME AND URL	PACKAGE	PHONE NUMBER	COST PER MONTH	MINIMUM CONTRACT TERM	CPU CORES / SPEED	DISK SPACE	1GBPS INTERNET CONNECTION	HARDWARE RAID	REMOTE POWER REBOOT	PERMANENT KVM	UPTIME GUARANTEE	NETWORK BACKUP STORAGE	PRIVATE SUBNET	24/7 PHONE SUPPORT
NETCETERA	2200DC	0800 8085450	£25	1 month	Dual Core 2.2GHz	160GB	N/A	Raid 1	✓	✓	✓	✓	✓	✓
	3000DC	0800 8085450	£40	1 month	Dual Core 3GHz	2 x 250GB	N/A	Raid 1	✓	✓	✓	✓	✓	✓
	2660QC	0800 8085450	£65	1 month	Intel 2.66GHz Quad Core Xeon Processor	2 x 500GB	N/A	Raid 1	✓	✓	✓	✓	✓	✓
	Developer	0800 8085450	£2.99	1 month	N/A	1GB	N/A	✓	✓	✓	✓	✓	✓	✓
	One	0800 8085450	£9.99	1 month	N/A	5GB	N/A	✓	✓	✓	✓	✓	✓	✓
Netcetera www.netcetera.co.uk/linux	Reseller	0800 8085450	£24.99	1 month	N/A	Unlimited	N/A	✓	✓	✓	✓	✓	✓	✓
Bravo14 (http://bravo14.co.uk)	Starter Linux	N/A	£20	N/A	N/A	2,000MB	N/A	✓	✓	✓	✓	✗	✓	✓
Bravo14 (http://bravo14.co.uk)	Starter Windows	N/A	£20	N/A	N/A	2,000MB	N/A	✓	✓	✓	✓	✗	✓	✓
Bravo14 (http://bravo14.co.uk)	Business Linux	N/A	£45	N/A	N/A	4,000MB	N/A	✓	✓	✓	✓	✗	✓	✓
Bravo14 (http://bravo14.co.uk)	Business Windows	N/A	£45	N/A	N/A	4,000MB	N/A	✓	✓	✓	✓	✗	✓	✓
Bravo14 (http://bravo14.co.uk)	Ultimate Linux	N/A	£60	N/A	N/A	Unlimited	N/A	✓	✓	✓	✓	✗	✓	✓
Bravo14 (http://bravo14.co.uk)	Ultimate Windows	N/A	£60	N/A	N/A	Unlimited	N/A	✓	✓	✓	✓	✗	✓	✓
catalyst2 (www.catalyst2.com)	Bronze Managed Dedicated Server	0800 107 79 79	£199	1 month	1x 2.4GHz vCPU	50GB	✓	✓	✓	✓	99.90%	✓	✓	✓
catalyst2 (www.catalyst2.com)	Silver Managed Dedicated Server	0800 107 79 79	£299	1 month	1x 2.4GHz vCPU	80GB	✓	✓	✓	✓	99.90%	✓	✓	✓
catalyst2 (www.catalyst2.com)	Gold Managed Dedicated Server	0800 107 79 79	£399	1 month	2x 2.4GHz vCPU	150GB	✓	✓	✓	✓	99.90%	✓	✓	✓
123-Reg (www.123-reg.co.uk)	Dell PowerEdge R200 (Ubuntu Linux)	0871 230 9525	£69.99	12 months	4x 2.13GHz	2x 160GB	10Mbit	✓	✓	✗	99.99%	O	✗	✓
123-Reg (www.123-reg.co.uk)	Dell PowerEdge R200 (Windows Web Edition)	0871 230 9525	£79.99	12 months	4x 2.13GHz	2x 160GB	10Mbit	✓	✓	✗	99.99%	O	✗	✓
Daily (www.daily.co.uk)	Linux VPS Pro	0845 466 2100	£29.99	1 month	2.27 Intel Quad Core	60GB	100Mbps	✓	✓	✗	✗*	✓ - full backup	✗	✗**
Daily (www.daily.co.uk)	Linux VPS Max	0845 466 2100	£59.99	1 month	2.27 Intel Quad Core	100GB	100Mbps	✓	✓	✗	✗*	✓ - full backup	✗	✗**
Daily (www.daily.co.uk)	Windows VPS Pro	0845 466 2100	£34.99	1 month	2.27 Intel Quad Core	60GB	100Mbps	✓	✓	✗	✗*	✓ - full backup	✗	✗**
Daily (www.daily.co.uk)	Windows VPS Max	0845 466 2100	£64.99	1 month	2.27 Intel Quad Core	100GB	100Mbps	✓	✓	✗	✗*	✓ - full backup	✗	✗**
Daily (www.daily.co.uk)	VPS Pro Hyper-V	0845 466 2100	£44.99	1 month	2.27 Intel Quad Core	60GB	100Mbps	✓	✓	✗	✗*	✓ - 1GB	✗	✗**
Daily (www.daily.co.uk)	VPS Max Hyper-V	0845 466 2100	£74.99	1 month	2.27 Intel Quad Core	100GB	100Mbps	✓	✓	✗	✗*	✓ - 1GB	✗	✗**
Daily (www.daily.co.uk)	VPS Ultra Hyper-V	0845 466 2100	£139.99	1 month	2.27 Intel Quad Core	200GB	100Mbps	✓	✓	✗	✗*	✓ - 1GB	✗	✗**
Heart Internet (www.heartinternet.co.uk/dedicated-servers)	Linux Dual Core	0845 644 7750	£79.99	12 months	Dual Core Xeon 2.33GHz	160GB	✓	✓	✓	✗	99.99%	✓	✗	24/7 Ticket support
Heart Internet (www.heartinternet.co.uk/dedicated-servers)	Windows Dual Core	0845 644 7750	£89.99	12 months	Dual Core Xeon 2.33GHz	160GB	✓	✓	✓	✗	99.99%	✓	✗	24/7 Ticket support
Heart Internet (www.heartinternet.co.uk/dedicated-servers)	Linux Quad Core	0845 644 7750	£129.99	12 months	Quad Core Xeon 2.5GHz	250GB	✓	✓	✓	✗	99.99%	✓	✗	24/7 Ticket support
Webfusion (www.webfusion.co.uk)	Dell PowerEdge R210	0845 130 1602	£79.99	12 months	2x 3.06GHz	250GB	Up to 100Mbit	✗	✓	✗	99.99%	Free	0	✓
Webfusion (www.webfusion.co.uk)	Dell PowerEdge R210	0845 130 1602	£119.99	12 months	4x 2.66GHz	2x 250GB	Up to 100Mbit	✓	✓	✗	99.99%	Free	0	✓
Webfusion (www.webfusion.co.uk)	Dell PowerEdge R210	0845 130 1602	£149.99	12 months	4x 2.66GHz	2x 500GB	Up to 100Mbit	✓	✓	✗	99.99%	Free	0	✓

O = Option

Dedicated and Shared server listings

	PACKAGE	PHONE NUMBER	COST	WEB SPACE	MONTHLY BANDWIDTH	POP3 ACCOUNTS	DATABASE SUPPORT	SHOPPING CART	VIRUS FILTER	FIREWALL	PHONE SUPPORT	EMAIL SUPPORT	WEB CONTROL PANEL	SERVICE LEVEL AGREEMENT
1&1 Internet Ltd www.1and1.co.uk	1&1 Starter (Linux)	0844 335 1211	From £0.99	10GB	Unlimited	10	✓	✓	✓	✓	✓	✓	✓	✗
	1&1 Basic (Linux)	0844 335 1211	From £0.99	100GB	Unlimited	100	✓	✓	✓	✓	✓	✓	✓	✗
	1&1 Basic (Win.)	0844 335 1211	From £2.99	100GB	Unlimited	100	✓	✗	✓	✓	✓	✓	✓	✗
	1&1 Unlimited (Linux)	0844 335 1211	From £4.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
	1&1 Unlimited (Win.)	0844 335 1211	From £4.99	Unlimited	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✗
	1&1 Unlimited Plus (Linux)	0844 335 1211	From £6.99	Unlimited	Unlimited	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗
	1&1 Unlimited Plus (Win.)	0844 335 1211	From £6.99	Unlimited	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✗
eHosting (www.ehosting.com)	Starter	0844 999 4100	£23.88	1GB	25GB	10	✗	✗	✗	✗	✗	✓	✓	✓
eHosting (www.ehosting.com)	Personal	0844 999 4100	£59.88	2.5GB	Unlimited	50	✓	✗	✗	✗	✗	✓	✓	✓
eHosting (www.ehosting.com)	Expert	0844 999 4100	£95.88	5GB	Unlimited	250	✓	✗	✗	✗	✗	✓	✓	✓
eHosting (www.ehosting.com)	Virtual	0844 999 4100	£227.88	50GB	Unlimited	Unlimited	✓	✗	✗	✗	✓	✓	✓	✓
Equiphase (www.equiphase.net)	Bronze	0121 314 4865	£30	200MB	2GB	10	✓	✓	✗	✓	✗	✓	✓	✓
Equiphase (www.equiphase.net)	Silver	0121 314 4865	£42	400MB	5GB	20	✓	✓	✗	✓	✗	✓	✓	✓
Equiphase (www.equiphase.net)	Gold	0121 314 4865	£72	800MB	10GB	100	✓	✓	✗	✓	✗	✓	✓	✓
Equiphase (www.equiphase.net)	Platinum	0121 314 4865	£114	1,200MB	40GB	200	✓	✓	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Email Only	02380 249 823	£40	1GB	2GB	10	✗	✗	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Essential	02380 249 823	£75	2GB	5GB	10	✗	✗	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Superior	02380 249 823	£140	5GB	10GB	25	✓	✓	✓	✓	✓	✓	✓	✓
Eurofasthost.com (www.eurofasthost.com)	Premium	02380 249 823	£250	10GB	25GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Starter	N/A	£29.99	500MB	1GB	3	✓	✓	✓	✓	✗	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Home	N/A	£54.99	2.5GB	30GB	50	✓	✓	✓	✓	✗	✓	✓	✓
Evohosting (www.evohosting.co.uk)	Business	N/A	£79.99	6.5GB	Unlimited	Unlimited	✓	✓	✓	✓	✗	✓	✓	✓
Evohosting (www.evohosting.co.uk)	eCommerce	N/A	£159.99	30GB	Unlimited	Unlimited	✓	✓	✓	✓	✗	✓	✓	✓
WebFusion (www.webfusion.co.uk)	Fusion Professional	0845 130 1602	£107.40	5GB	50GB	1,000	✓	✗	✓	✓	✓	✓	✓	✗
WebFusion (www.webfusion.co.uk)	Fusion Business	0845 130 1602	£179.40	10GB	150GB	1,500	✓	✗	✓	✓	✓	✓	✓	✗
WebFusion (www.webfusion.co.uk)	Fusion Developer	0845 130 1602	£227.40	20GB	300GB	5,000	✓	✗	✓	✓	✓	✓	✓	✗
WebFusion (www.webfusion.co.uk)	Fusion Reseller	0845 130 1602	£329.99	Unlimited	Unlimited	Unlimited	✓	✗	✓	✓	✓	✓	✓	✗
Giacom (www.giacom.com)	Business Pro	0800 542 7500	£199	100MB	2GB	100	✓	✓	✓	✓	✓	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Starter Professional	0845 644 7750	£29.80	2.5GB	10GB	1,000	✗	✗	✗	✓	✗	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Home Professional	0845 644 7750	£89.99	10GB	50GB	10,000	✓	✓	✗	✓	✗	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Business Professional	0845 644 7750	£129.99	Unlimited	Unlimited	Unlimited	✓	✓	✗	✓	✗	✓	✓	✓
Heart Internet (www.heartinternet.co.uk)	Reseller Professional	0845 644 7750	£299.99	Unlimited	Unlimited	Unlimited	✓	✓	✗	✓	✗	✓	✓	✓
Hostway (www.hostway.co.uk)	Silver	0808 180 1880	£79.50	150MB	3GB	5	✗	0	✓	✓	✗	✓	✓	✗
NameHog (www.namehog.net)	Email Only	0845 612 0330	£11.75	25MB	1GB	5	✗	✗	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Standard Package	0845 612 0330	£35.25	100MB	4.5GB	10	✗	✗	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Professional Package	0845 612 0330	£58.75	250MB	8GB	25	✓	✓	✓	✓	✓	✓	✓	✓
NameHog (www.namehog.net)	Expert Package	0845 612 0330	£105.75	500MB	15GB	75	✓	✓	✓	✓	✓	✓	✓	✓
Skymarket (www.skymarket.co.uk)	Standard 1	0800 321 7788	£49	10MB	2GB	1	✓	✗	✓	✓	✓	✓	✓	✓
Skymarket (www.skymarket.co.uk)	Standard 2	0800 321 7788	£69	20MB	2GB	1	✓	✗	✓	✓	✓	✓	✓	✓
Skymarket (www.skymarket.co.uk)	Premium 1	0800 321 7788	£99	25MB	2GB	1	✓	✗	✓	✓	✓	✓	✓	✓
Blacknight (www.blacknight.com)	Minimus Linux / Windows	+44 (0)845 5280242	€49.95	10GB	150GB	1,500	✓	✓	✓	✓	✓	✓	✓	✗
Blacknight (www.blacknight.com)	Medius Linux / Windows	+44 (0)845 5280242	€89.95	20GB	300GB	5,000	✓	✓	✓	✓	✓	✓	✓	✗
Blacknight (www.blacknight.com)	Maximus Linux / Windows	+44 (0)845 5280242	€149.95	30GB	600GB	Unlimited	✓	✓	✓	✓	✓	✓	✓	✗

O = Option

YOUR VIEW

Linux User Letters

Your opinions about the magazine, Linux and open source

Alternative Pi

I was very interested to read the review by Gareth Halfacree in Issue 139 of **Linux User & Developer** on the Cubieboard2 and, like many other readers I am sure who were partly encouraged by the '4 penguins' rating, I decided to go ahead and purchase one.

What I would be very interested to learn, though, is just how many other readers were similarly encouraged to buy one and, if so, how they are getting on with it? As Gareth mentions in his review, current support for this little device is not that easy to track down, as some of the best developers have now moved away from the 'official' support forum, which is a shame.

One of the most active of these was a Spanish contributor 'ikeeki', who helped to produce the excellent Qbee-X image, which can be found on the Cubie Forums and is probably one of the 'best' currently available.

I have been evaluating it over the last few days and am just waiting for the Cubieboard2 version to be tweaked a bit more before it is officially launched. Hopefully **Linux User & Developer** will be publishing more articles on this type of 'alternative' board.

Thanks again for a great magazine, to which I have just subscribed!

David Rolfe

You're actually the first one to write in about it, but you may not be the last! As Gareth said in his review, it's a logical step after the Raspberry Pi and we do recommend it if you find yourself needing just that little bit more. We'll certainly consider doing more content based on it in the future if it's appropriate.

XPired

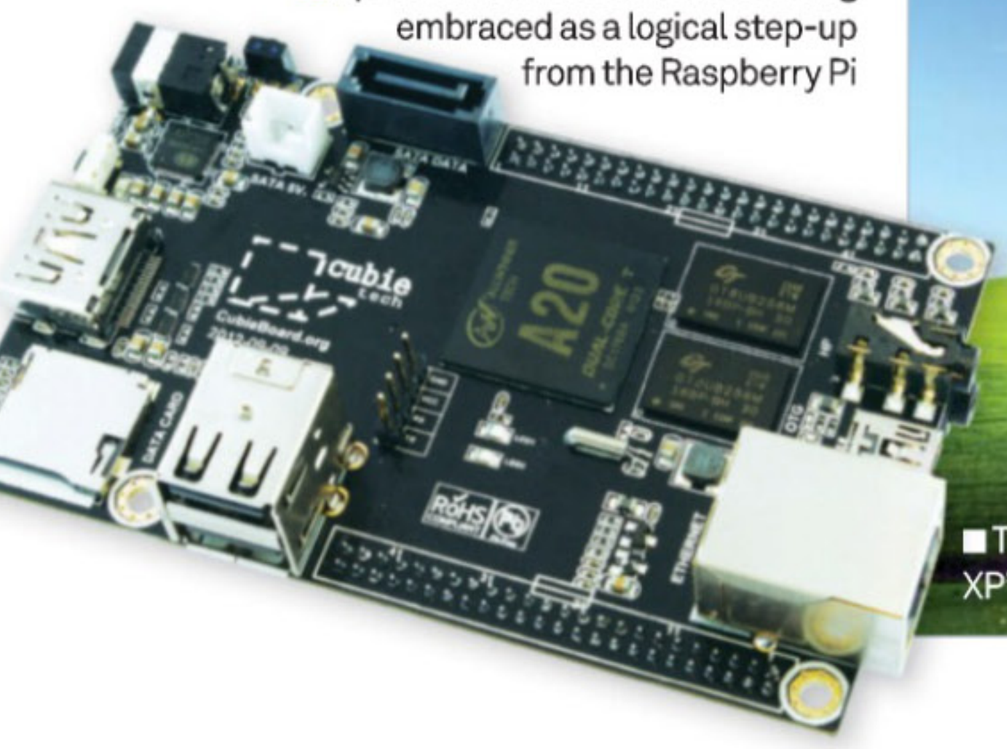
It's remarkable that proprietary software, while so popular, has the tendency of becoming obsolete and buggy after a few years with a difficult upgrade path. Wouldn't it be marvellous if there were an alternative that you could easily update and never need to upgrade? It amazes me to this day that Linux distros for desktop aren't as popular as Windows or OS X for this very reason. Hopefully this Windows XP change will get more users on Linux and they'll find out they won't 'need' to change ever again.

Michael Moss

Desktop Linux still has a bit of stigma, unfortunately, and computing skills among people aren't the best either. The irony is most – if not all – of them will be using Linux in some form in their day-to-day lives without even knowing; whether it's Android, a web server somewhere along the line, or even integrated into their car. We've decided to do a full feature on switching to Linux this very issue, so if there are any people in your life looking to make the switch, make sure you point them our way!

Remaster with Refracta

I read your 'Build your own distro' article with great relish a few issues back, and was sad to be reminded that Remastersys has not really had much active development for years now. However, there is a new system that might be good for those looking for a replacement:



■ A powerful little board that is being embraced as a logical step-up from the Raspberry Pi

■ The grass is greener on the other side of XP's most famous background

Twitter:
@linuxusermag

Facebook:
Linux User & Developer

Refracta. It's based on Debian and it allows you to create a very custom version of that to then use on a live CD or USB storage. Will we see any tutorials on this and other available options in the future?
Terrence Moore

Yes, there haven't seemed to be any major forks of Remastersys just yet and Refracta could well be the successor to it of sorts – we'll definitely consider using it in the future. While we're talking about alternatives, there is also *Linux From Scratch*, which guides you to compiling a kernel and relevant software entirely from nothing. It's for much more advanced users but give it a shot if you want a challenge.

Android Magazine

Since there are more and more Android features that are being merged back into the Linux Kernel, it's becoming easier to develop for one of the most popular mobile operating systems around. With over 25 billion app downloads and over 100 million Android devices worldwide, there's a wide audience ready and willing to consume apps. For a more Android-driven editorial, you can look to our sister mag **Android Magazine**, the only publication dedicated to the platform. Along with news and reviews to keep you up to date on everything Android, there are also tutorials and advice on developing and hacking your hardware. Find out more at **www.littlegreenrobot.co.uk**

THREAD BARE

Of GNU and Linux

Richard Hillesley talked about the relationship between Red Hat and CentOS in one of his free software columns, highlighting the competition issues faced in traditional business environments simply by CentOS' existence.

Our community had their own thoughts on this and his use of 'GNU/Linux'. Check it out at: **http://bit.ly/1pM3dyX**

jgmiller said:

How could it have been a purchase, when on the Red Hat Community blog at Red Hat, employee Karsten Wade makes it abundantly clear that:

"CentOS is a community project not a legal entity, so there is nothing to purchase."

Reading further there, perhaps the primary reason for Red Hat to embrace CentOS is Openstack:

"Another is that code in projects such as OpenStack is evolving without the benefit of spending a lot of cycles in Fedora, so our projects aren't getting the community interaction and testing that the Linux base platform gets."

So the commercial reason (and commercial reasons are why corporations are motivated to do anything) appears to be all about getting a low cost 'platform' to test out OpenStack and other Red Hat projects which are business server oriented and thus more relevant to CentOS users, rather than having relevance to Fedora's predominantly desktop/laptop-oriented user base.

Cory Hilliard said:

I think the use of GNU when referring to Linux also needs to be stopped. GNU is a project that is owned by Richard Stallman. He himself says that Linux is not part of GNU and Linux doesn't have the same direction or views as GNU does. Linux is built by Linus Torvalds with its own goals and direction. If you were a part of GNU then yes, making the distinction that it is GNU/Linux is important, because you aren't using GNU/Hurd, and therefore need to make sure people know that the GNU/Linux operating systems weren't free in the same way that RMS wants them to be. So Linux is Linux (the OS) and GNU/Linux is something that Richard is using because Hurd isn't as good as Linux (the Kernel).

mike said:

I dare you to say that in front of a computer science class. An OS is, by definition, a kernel (Linux) + user space programs (GNU). You can have a Linux-based OS without GNU (as Android) or a GNU-based OS without Linux (GNU/Hurd, GNU/kFreeBSD). Linus is free to dislike Stallman's view, but he is technically wrong.

Jeffrey Tee said:

Referring to Linux as GNU/Linux gives recognition to the fact that the Linux OS is built up of Linux (the kernel) and a lot of the GNU Projects tools.

The 'Linux' OS is not part of GNU, but contains tools developed as part of the GNU Project. I don't think you need to refer to the GNU OS (with the Hurd kernel) as GNU/Hurd as the Hurd is part of the GNU project, not a combination of two projects.

Richard Stallman objects to the pragmatic direction the Linux Kernel takes as regards allowing non-free software and the kernel in the matter of proprietary drivers for video/wireless cards, etc. The Kernel and most of the other tools that make up the Linux system share the same software licence (the GPL), only additional tools such as Apache, Tmux, some office packages have different licences.


Explore
Read
Shop
SAMSUNG
greatdigitalmags - The official download website of Imagine Publish
HOME
SUBSCRIPTIONS
CONTACT US
Now enjoy yo
mag howeve
Magazines
eBooks
Apps
27
Tweet
1
+1
360 Magazine
3D Artist
✓iPad ✓iPhone ✓Android phone ✓Android tablet ✓Apple Mac ✓Windows PC

No Disc. No Problem

Many of the files you're looking for can be found on the magazine's website

Imagine digital editions are a new and exciting way to experience our world-leading magazines and bookazines.

To get the most out of your digital editions, be sure to enjoy all of our fantastic features, including:

- Zoomable text and pictures
- In-app browsing
- Searchable text
- Take your collection with you
- Read offline

Advanced Photoshop

Android Magazine

To buy more Imagine digital editions and for the latest issues and best offers, please go to **www.GreatDigitalMags.com**

Next issue

What's coming up...

Look out for issue 142 on sale 31 July
Want it sooner?
Subscribe today!

COMING IN THE NEXT ISSUE

LINUX MINT 17 QIANA

Linux Mint comes of age with Qiana – but is it enough to make it the world's best distro?

Raspberry Pi games console

Rob shows us how to build a Pi-powered, hand-held retro gaming machine

Learn computer science for free

Find out how MOOCs are changing education

How to write Bash scripts - part 1

The first in a three-part series, next month we'll be learning how to automate tasks in Linux

ALSO INSIDE:

» LXLE: lightweight Linux reviewed
» Best web browsers: group tested
» Manage your projects with LibrePlan
» Code Android apps with Java – part 2

Linux User & Developer
THE MAGAZINE FOR THE GNU GENERATION

Imagine Publishing Ltd
Richmond House, 33 Richmond Hill
Bournemouth, Dorset, BH2 6EZ
☎ +44 (0) 1202 586200
Web: www.imagine-publishing.co.uk
www.linuxuser.co.uk
www.greatdigitalmags.com

Magazine team

Deputy Editor Gavin Thomas
gavin.thomas@imagine-publishing.co.uk
☎ 01202 586272
Production Editor Victoria Richards
Designer Sam Ribbits
Senior Staff Writer Rob Zwetsloot
Photographer James Sheppard
Senior Art Editor Will Shum
Editor in Chief Nick Roberts
Publishing Director Aaron Asadi
Head of Design Ross Andrews

Contributors

Joey Bernard, Simon Brew, David Crookes, Gareth Halfacree, Tam Hanna, Richard Hillesley, Jon Masters, Richard Smedley, Sean Tracey, Nitish Tiwari, Mihalis Tsoukalos.

Advertising

Digital or printed media packs are available on request.
Head of Sales Hang Deretz
☎ 01202 586442
hang.deretz@imagine-publishing.co.uk
Advertising Manager Alex Carnegie
☎ 01202 586430
alex.carnegie@imagine-publishing.co.uk
Account Manager Rhian Carter
☎ 01202 586421
rhian.carter@imagine-publishing.co.uk

Cover disc

Junior Web Designer Steve Dacombe
ludxtrahelp@imagine-publishing.co.uk

International

Linux User & Developer is available for licensing.
Head of International Licensing Cathy Blackman
☎ +44 (0) 1202 586401
licensing@imagine-publishing.co.uk

Subscriptions

For all subscriptions enquiries
☎ 0844 249 0282 (UK)
☎ +44 (0) 1795 418661 (Overseas)
Email: LUD@servicehelpline.co.uk
6 issue subscription (UK) – £25.15
13 issue subscription (Europe) – £70 (ROW) - £80

Circulation

Head of Circulation Darren Pearce
☎ 01202 586200

Production

Production Director Jane Hawkins
☎ 01202 586200

Founders

Group Managing Director Damian Butt
Group Finance & Commercial Director Steven Boyd

Printing & Distribution

Printed by William Gibbons, 26 Planetary Road, Willenhall, West Midlands, WV13 3XT

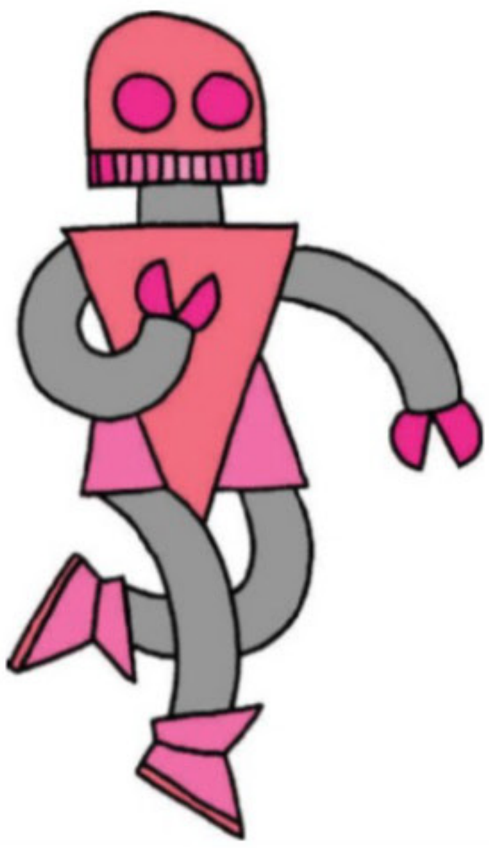
Distributed in the UK, Eire & the Rest of the World by:
Marketforce, Blue Fin Building, 110 Southwark Street
London, SE1 0SU
☎ 0203 148 3300
www.marketforce.co.uk

Distributed in Australia by:
Network Services (a division of Bauer Media Group)
Level 21 Civic Tower, 66-68 Goulburn Street
Sydney, New South Wales 2000, Australia
☎ +61 2 8667 5288

ISSN 2041-3270

IMAGINE PUBLISHING

recycle
When you have finished with this magazine please recycle it.



Can you volunteer for Code Club?

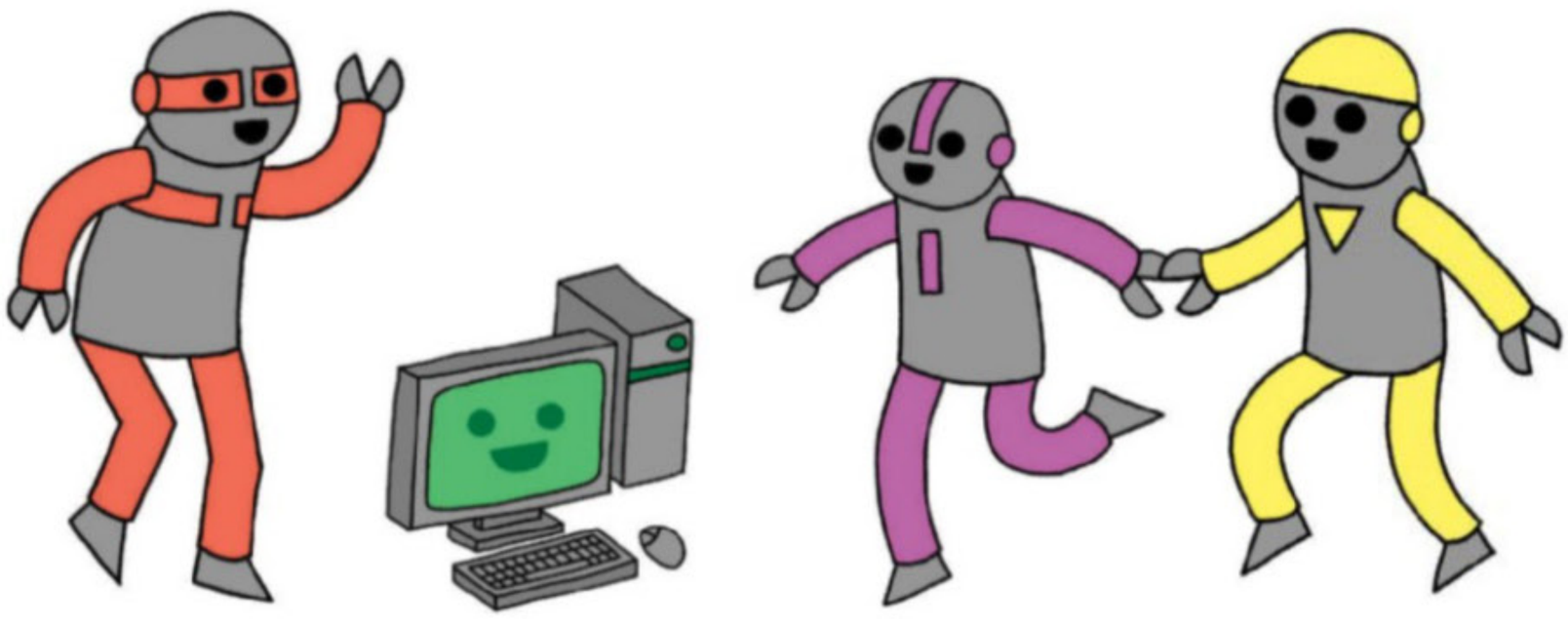
Code Club is a nationwide network of volunteer-led after school coding clubs for children aged 9-11.

We need people who know how to program computers to volunteer to run a club at their local primary school, library or community centre for an hour a week.

We create the projects for our volunteers to teach, the projects we make teach children how to program by showing them how to make computer games, animations and websites.

Get involved, let's teach the next generation to code!

Visit **www.codeclub.org.uk** to find out more



poundhost
A NAMESCO COMPANY

24/7 UK SUPPORT

Dedicated Servers

intel SSD

NEW DC

33% OFF

Quote Voucher code:
PHLUD33

- No Contracts
- 24/7 OnSite Support
- Dell Hardware
- Varied Server Range

From
£19 per month

Dedicated Servers | Cloud Servers | VPS | Domains | Email | Hosting | SiteMaker | Ecommerce | Servers | SSL | Site Promotion

Dada Figures and Stats:

490,000 Number of Customers

8,000 Number of Dedicated Servers

6,000 Number of Virtual Servers (VPS)

1,800,000 Number of Domains

tel: **0800 876 0942**
www.poundhost.com

poundhost
A NAMESCO COMPANY