UBUNTU 23.10
Explore the features of Mantic Minotaur

INSIDE POP!_OS
All the secrets of the new Rust desktop

HOME WEBSITES
WordPress turns 20 so fire up a web server!

LINUX FORMAT

The #1 open source mag

Inside the all-new Raspberry Pi 5!

STREAMING MADE SIMPLE!

Release your media with open source Jellyfin and enjoy it wherever and whenever you want!

+ STREAM FREE TV
page 62

PLUS: HOW TO

- Find files faster using the terminal
- Edit and enhance your photos with Fotoxx
- Stream and play games on your Raspberry Pi

COBOL LIVES!
The least liked language still powers mainframes

FREE PASCAL
The easy way to code Linux GUI apps

INSIDE NETWORKS
Explore how packets get about your cables

LXF November 2023
FUTURE

LINUX FORMAT

MEET THE TEAM

This issue, we're setting up a *Jellyfin* media server – that was Neil's first ever Linux-based project. What first got you installing Linux back in the day?

Matthew Holder
A friend convinced me to install Slackware on a spare PC to learn about Linux. I then decided to share the pain of dial-up internet across multiple devices – imagine the speed shared between three machines! I haven't looked back or stopped learning since the year 2000.

Les Pounder
A magazine coverdisc in the late '90s. Not sure of the exact distro but it was RPM-based. It got me interested enough in Linux to download Corel Linux, then Mandrake. I later switched to SUSE, then distro hopped for a while before settling with Ubuntu.

Mayank Sharma
I first installed Linux back when installing Linux was a project in itself. My dad read about the Halloween Documents and ordered the official Slackware CD set. To install Linux back then you really had to RTFM, and fail at least thrice before getting it right.

Nick Peers
The earliest experience with Linux I can remember wasn't with a desktop PC, but with an ageing router. Faced with an expensive upgrade, I discovered the brilliant DD-WRT in 2004, a Linux-based OS that upgraded its security and added new features to boot.

David Rutland
I'd been aware of Linux for a while, but what really got me interested was the publication of the Halloween Documents. If Linux and open source were good enough to seriously frighten Microsoft, it was something I wanted to try myself. Two decades later, I'm still here!

At your service

I recently expanded my home server's RAID to 20TB, adding a fifth 4TB drive. It's an ongoing upgrade process that's probably been going on for a decade in one form or another, and running Linux ensures it's always super-smooth. The only recent hiccup was that the Nvidia GPU driver started causing suspend issues out of the blue. So, I fell back to using the AMD Athlon 200G APU, which was surprisingly capable at transcoding even 4K video.

This was my original 'in' to Linux: setting up a home file and media server. Back when Microsoft was getting draconian with licences, it just seemed simpler dealing with a Linux distro that you can install where, when and however you like, without bundled bloat and extra hurdles involved.

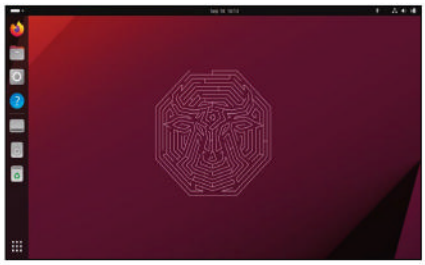
So, if you're picking up this magazine and fancy giving Linux a try, creating a home media server is the perfect first project. Older systems are perfectly capable of streaming 4K content to your TV, never mind everything else we're looking at, from audiobooks to photos. While you're at it, if there's enough storage, you're also getting a file server, print server and anything else you fancy chucked in for free, too.

The sheer versatility of Linux is the real winner for me. This issue alone, it's powering our media, sorting our photos, exploring coding old and new, reviving the classic Domesday Project and streaming modern games, all for us to enjoy!

Neil Mohr Editor
neil.mohr@futurenet.com

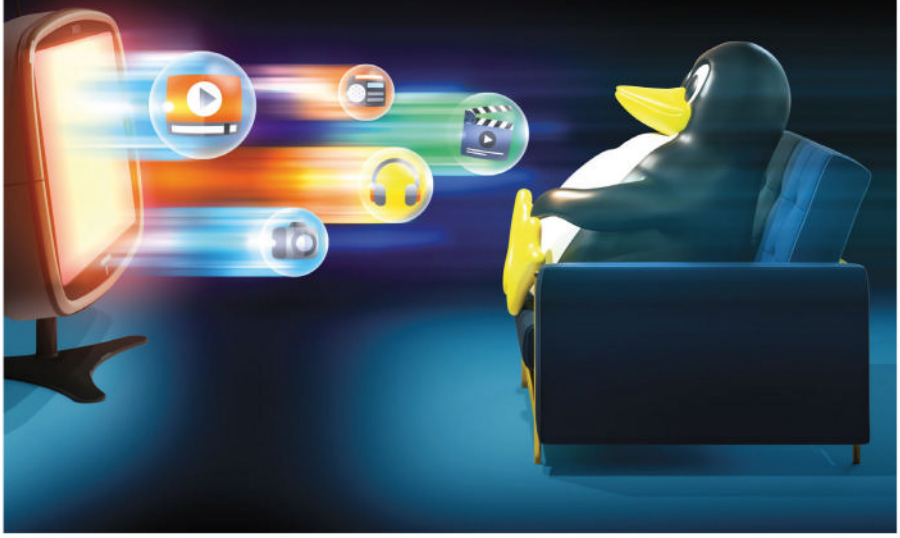
Contents

REVIEWS



STREAMING MADE EASY!

Bring together all your movie, TV, music, audiobook and photo streaming needs under one roof. Nick Peers shows you how.
See page 34!



CREDIT: Magictorch

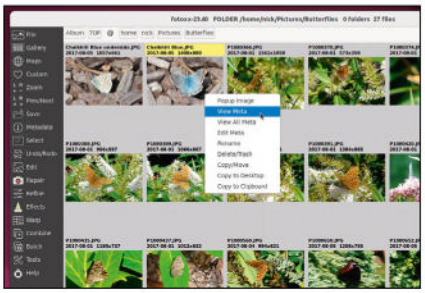
ROUNDUP

IN DEPTH

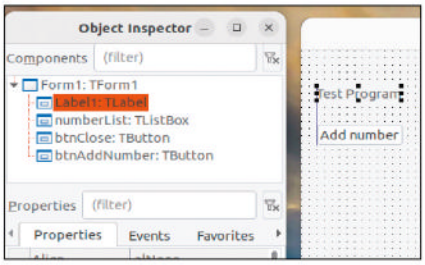
CREDIT: iStock/Getty Images Plus/Rawpixel

CONTENTS

TUTORIALS

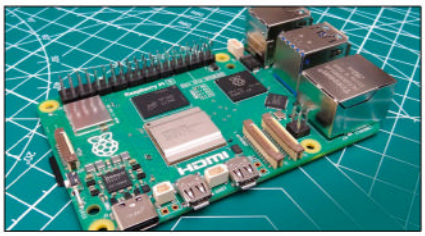


CODING ACADEMY

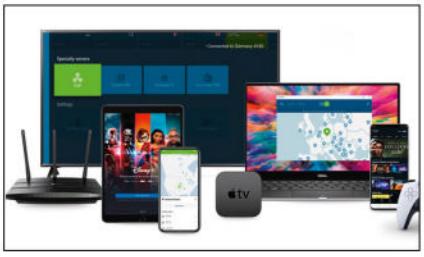


REGULARS AT A GLANCE

Pi USER



ADMINISTERIA



Newsdesk

THIS ISSUE: GNU turns 40 » Gnome hits 45 » OpenTofu looks tasty » LibreOffice downloads soar » Kernel embraces AV1

GNU IS NOT UNIX

Free Software Foundation celebrates GNU at 40

Since 1983, GNU has formed the backbone of key computer systems. The FSF reflects on all that's great about copyleft and free software.

Remember those annoying people who tell you it's not called Linux, but GNU/Linux? This dates back to May 1983 when Richard Stallman used the Stanford bulletin board to announce plans to develop a free Unix-like operating system called GNU. The name itself is a recursive acronym for "GNU's not Unix". Stallman's philosophy of free software scored a very deep line in the sand between visionaries like himself and private companies producing proprietary programs.

In 1985, Richard founded the non-profit Free Software Foundation (FSF) to support the free software movement. It continued existing GNU projects and also hired developers to produce more free software programs.

The FSF also pioneered the concept of copyleft, whereby software licensing requires the same rights to be preserved in derivative works. As Stallman stated, "The goal of GNU was to give users freedom, not just to be popular. So we needed to use distribution terms that would prevent GNU software from being turned into proprietary software."

This culminated in the GPL (GNU Public License), which is widely used for free software.

In 1992, the Linux kernel was rereleased under the GPL, making it free software. The combination of GNU and Linux formed a free operating system, which made it possible for the first time to run a PC without non-free software. As most developers know, most modern Linux distros do include some proprietary code, such as firmware. Still, the FSF maintains a list of approved Linux distros that use entirely free software, such as Trisquel and Dragora.

Since 2009, the FSF has also held the annual LibrePlanet conference to discuss software freedom and digital ethics. These include a welcome speech from Stallman himself, as well as contributions from guest speakers, such as Edward Snowden in 2016.

On 27th September, GNUnet celebrated GNU's 40th anniversary with a hacker meeting in Switzerland, featuring presentations about various GNU packages, hacking and making new releases. The FSF also held a hackday on 1st October at its offices in Boston, Massachusetts, for families, students and everyone else interested in both hacking and celebrating GNU's anniversary.

Zoë Kooyman, FSF executive director, said: "The GNU system isn't just the most widely used operating system based on free software. It's also at the core of a philosophy that has guided the free software movement for 40 years."

During Richard Stallman's closing speech at the GNU 40th anniversary event in Switzerland, he revealed that he is undergoing treatment for non-Hodgkin lymphoma, a form of cancer of the white blood cells. Appearing on stage, Stallman lacked his typical characteristic long hair and beard due to his treatment, which he says is progressing well and that he still hopes to be around and active in GNU for years to come.

Visit **https://fsf.org** and **www.gnu.org/gnu40/** to learn more about GNU and the FSF's ethos and to see Stallman's recorded speech.

'Escape to Freedom' is a new animated video from the FSF, giving an introduction to the concepts behind software freedom.

The GNU.org website has announced a number of events to celebrate the 40th anniversary of GNU, including hackers' meetings.

DESKTOP ENVIRONMENT

Gnome 45 is coming

Gnome 45 'Rīga' focuses on improved performance plus overhauling windows and file management.

On 20th September, the Gnome Project was proud to announce the long-awaited release of Gnome 45, code-named Rīga in honour of the organisers of this year's Gnome community conference.

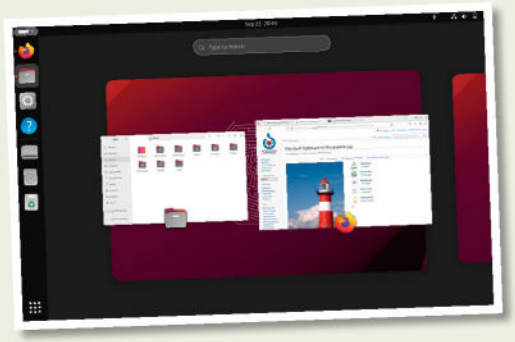
We were treated to a preview of all Gnome 45 has to offer while testing daily builds of Ubuntu 23.10 *(see review, p.24)*. The biggest change to the desktop is that the Activities button is no more – instead there's a small workspace indicator. Handily, it's dynamic in that it indicates which workspace you're currently using.

The search feature has also been overhauled. Not only are searches faster and less resource-intensive, but the *Nautilus* file manager now enables users to run global searches, not just within their home folder. The searches themselves can now be ordered in a more useful way, plus there are helpful buttons such as Search Everywhere to expand search scope.

The default image viewer app has been replaced with a slick alternative that supports gestures on a touchscreen or touchpad. There's also a new properties sidebar to copy, trash or print images. The image viewer is optimised for both mobile and desktop devices, as is the new camera app, which includes overlaid controls in the viewing area.

While we're talking apps, a number of core programs have an overhauled UI. Sidebars now run to the full height of windows in apps such as *Files, Calendars, Settings* and *Contacts*. There are numerous other style updates, such as an updated header bar style for programs such as *Console* and the *Disk Usage Analyzer*.

Gnome 45 also boasts two brilliant new wallpapers, as well as improved keyboard shortcuts and pointer visuals. Read the release notes at **https://release.gnome.org/45/**.



Gnome 45 replaces the Activities button with a handy dynamic workspace switcher.

SOFTWARE

From Terraform to OpenTofu

OpenTofu is now overseen by the Linux Foundation.

Back in August, HashiCorp, the vendor of deployment-automation tool *Terraform*, changed its software licence from MPL (Mozilla Public Licence) to BUSL (Business Source Licence). This means it is technically no longer FOSS. Crucially, the new licence forbids users from using source code to develop competing products.

Shortly after, a splinter group announced that it was launching the OpenTF Project and released the official open source fork of *Terraform, OpenTF*. In September, the group renewed its commitment to FOSS by placing the project under the oversight of the Linux Foundation as *OpenTofu* – the fork's new official name. The project members have also announced that the project will soon be submitted for entry into the CNCF (Cloud Native Computing Foundation).

Yevgeniy (Jim) Brikman, a key member of the *OpenTofu* team, says the name was deliberately playful. (Though we can't help but feel they missed a trick by not naming it *TerraFork.)*

Brikman also claims that key players such as Google, Microsoft and Amazon are sitting on the fence. Currently *OpenTofu* is a drop-in replacement, as it's compatible with *Terraform* versions 1.5.x. However, the *OpenTofu* website says that future compatibility and features will be decided by the community. The site also claims that some companies have pledged to pay for full-time engineers to work on *OpenTofu* and that 19 are involved already. The alpha release is available via **https://opentofu.org**.

CREDIT: gnu.org

OPINION

TOFU FOR ALL!

Joe Brockmeier
is head of community, Percona.

"You can read about HashiCorp's licence rug-pull below-left. The community, after a few warning shots, forked *Terraform* as *OpenTF*. At the Open Source Summit Europe, the Linux Foundation (LF) validated and endorsed the project by putting it under the foundation's umbrella as *OpenTofu*.

This is important for two reasons. First, *OpenTF* would probably have succeeded without the LF's backing. But endorsing the project and providing a foundation home (under a name less likely to be a trademark issue) all but guaranteed its success.

This is important, since the LF's member base and many projects under the LF's Cloud Native Computing Foundation (CNCF) depend on and were designed with *Terraform* in mind. This allows the ecosystem to move forward, on a technical level, with minimal disruption.

Secondly, this is a political statement. As more and more businesses flirt with relicensing under non-open source licences, the LF showed willingness to back forks to protect the larger industry. Abrupt relicensing of open source projects is harmful. It undermines decades of work in building goodwill and trust in open source licensing. Companies that abuse the community's goodwill have been served notice that it won't be tolerated."

OPINION

CODEC COUP

Benjamin Gaignard
is a senior software engineer at Collabora.

Released in August, Linux kernel 6.5 brought 22 patches that enabled support for the AV1 uAPI and for two stateless video decoders: one for the Rockchip RK3588 and one for MT8195, a MediaTek SoC. This is the result of several years of efforts by our team to bring up this new codec for stateless video decoding.

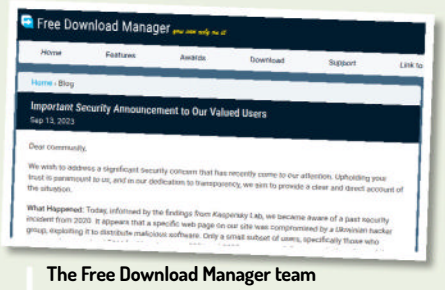
Adding a new video codec is not simple work because it requires a stable uAPI and at least two drivers using it on different hardware. The two drivers are needed because they help prove that the uAPI can meet the various needs for hardware from different vendors. As a matter of fact, having more drivers would have been even better in terms of testing and coverage, but there are currently only two hardware IP blocks known to include a stateless AV1 decoder.

Enabling AV1 support heavily benefits the open source community, as AV1 itself is an open source codec. This means that AV1 has a royalty-free licensing model that makes it suitable for adoption in various open source projects. With AV1 hardware designs starting to become more common in consumer devices, users can now better leverage the capabilities of the hardware they have paid for while using Linux.

MALWARE

Download manager was trojaned

Team issues script to check if Linux systems have been compromised.



The Free Download Manager team explained the link to malware was inadvertently removed via an update.

Download managers can be an excellent way to juggle multiple file transfers in parallel, but in the case of the *Free Download Manager* for Linux, it's been a surefire way to infect systems.

While *Free Download Manager* itself is a legitimate app, in September Kaspersky researchers discovered that the developers' website had been compromised. Between 2020 and 2022, users were being redirected to a dodgy Debian repository containing a malicious version of the program. Once installed, the fake download manager tampers with *Cron* to make sure it starts up each time users log in to the Linux operating system. It then launches a reverse shell, which it uses to collect sensitive data, such as system information, browsing history, saved passwords and cryptocurrency wallet files, as well as credentials for cloud services.

The FDM team issued a statement in response to the findings, apologising to users and explaining the compromised links were fixed in 2023. The team is also offering a Bash script to detect the malware: **www.freedownloadmanager.org/blog/?p=664**.

SOFTWARE

Millions of LibreOffice!

The LibreOffice suite soars in popularity.

The LibreOffice Document Foundation can rightly feel proud of the fact that there have been over 1.5 million downloads of *LibreOffice 7.6*. The number of users is probably much higher, as many versions of Linux come with the office suite preinstalled.

New features include a very welcome wizard to insert page numbers into *Writer*, better support for drawing styles in *Calc*, and a navigation panel for switching slides during presentations for *Impress*. Find out more: **https://bit.ly/lxf308libre**.

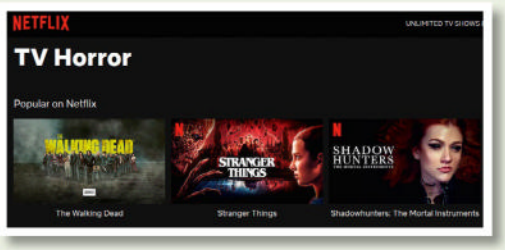
The LibreOffice Document Foundation posted about record downloads, along with a video showcasing new features.

CODECS

AV1 decoder

Kernel support for AV1 uAPI and stateless video decoders.

Since 2021, the Linux kernel has been preparing to support encoding/decoding the royalty-free AV1 codec. It began with a "request for comments" patch series by Collabora's Daniel Almeida for implementing the stateless AV1 user-space API within the media subsystem. The first uAPI tentatives needed seven versions on the mailing list before finally being considered stable in June 2023. The latest Linux kernel (v6.5) includes patches that enable support for AV1 uAPI as well as two stateless video decoders: one for the Rockchip RK3588 and one for MT8195, a MediaTek SoC. Find out more (*see left*) **https://bit.ly/lxf308av1**.



The AV1 codec is royalty-free and highly efficient. Since 2022, Netflix has used it for streaming selected titles.

CREDIT: https://blog.documentfoundation.org/. https://www.freedownloadmanager.org/blog/?p=664. www.netflix.com

Distro watch

What's behind the free software sofa?

FREEBSD 13.2

Linux Format is the home of open source, including alternative -nix operating systems, such as FreeBSD. Although the OS is a popular choice for servers, it supports a number of desktop environments and window managers for those willing to manually install the packages. The latest stable release includes better support for installing on SD cards, updated versions of OpenSSH and OpenSSL, and integration for VPN interface using WireGuard. Read more at **www.freebsd.org**.

FreeBSD code is used in OS X and the PlayStation 4.

CACHYOS 20230917

CachyOS was first released in late 2022. It's based on Arch Linux with a heavily optimised kernel BORE (Burst-Oriented Response Enhancer) scheduler. The distro has its own repositories, and desktop packages are compiled with LTO and x86-64-v3 optimisation, security flags and performance improvements, especially for older Nvidia and AMD cards. Downloads are available for multiple desktop environments including Plasma and Gnome. The OS also supports multiple boot environments and has its own *Firefox*-based browser. **https://cachyos.org/download**

CachyOS is designed for optimised performance.

KAOS 23.09

Kaos describes itself as a "lean KDE distribution". Although it's inspired by Arch, the operating system is built from scratch and maintains its own package repositories (managed via *Pacman).* Besides using Plasma 5.27.8, it also bundles the *Qt* toolkit, as well as the *LibreOffice* suite. The current build is based on v6.4.16 of the Linux kernel. The KaOS team has also recently released a 'preview' ISO using the Plasma 6 desktop. **https://kaosx.us/news/2023/kaos09**.

Kaos is Arch-inspired and comes with the KDE Plasma desktop.

MANJARO 23.0.2

Manjaro Linux is a fast, desktop-orientated OS based on Arch Linux. Its developers claim that the distribution is designed to be user-friendly. The fact that it follows a stable rolling release model is good proof of this, as well as the fact that it uses Pacma, a graphical front-end for *Pacman.* Installation is handled by the ever-intuitive *Calamares.* The latest Manjaro release 'Uranos' is available with Gnome 44, Plasma 5.27 LTS or the Xfce 4.18 desktop environments.

Manjaro uses the Calamares installer and Pacma.

FATDOG64 900

Based on Puppy Linux, this is a 64-bit desktop build, which as the name suggests, is a somewhat more porky install than the original Puppy iteration. Version 900 is based on Linux From Scratch 11.3 (replacing LFS 8.2 that the 800 series was using), and the release notes point out that this has initial HiDPI support with Kernel 6.1.46. For full details, rush to **https://distro.ibiblio.org/fatdog/web/900.html**.

The RSPCA might take a dim view of this canine's diet.

OPINION

SECURITY FIRST

Italo Vignoli
is one of the LibreOffice and Document Foundation founders.

"The Document Foundation is releasing *LibreOffice 7.6.2 Community* and *LibreOffice 7.5.7 Community* ahead of schedule to address a security issue known as CVE 2023-4863, which originates in a widely used code library known as libwebp, created by Google more than a decade ago to render the then-new WebP graphics format.

The new versions also fix other bugs and regressions and are available immediately from **www.libreoffice.org/download**. All users of *LibreOffice* are encouraged to update their current version as soon as possible.

For enterprise-class deployments, the Document Foundation strongly recommends the *LibreOffice Enterprise* family of applications from ecosystem partners – for desktop, mobile and cloud – with a wide range of dedicated value-added features and other benefits such as SLAs: **https://bit.ly/lxf308lbinb**.

The Document Foundation does not provide technical support to users, although it is available from volunteers on user mailing lists and the Ask LibreOffice website: **https://ask.libreoffice.org**.

LibreOffice users, free software advocates and community members can support the Document Foundation by making a donation at **www.libreoffice.org/donate**."

OPINION

MAC MAGIC

Jon Masters is a kernel hacker who's been involved with Linux for over 22 years, and works on energy-efficient Arm servers.

"I've started digging into the Asahi Linux port to Apple silicon. This is a multi-year effort to bring Linux to M-series Macs. They can already boot Linux kernels, support most of the hardware, and provide a good desktop experience. The underlying hardware is famous for being blazingly fast, so it is a joy to watch a compilation as the object files fly by on the screen.

One of the best things about the project is that it is focused on getting supportable patches into an upstreamable state. For example, Asahi boots using a special bootloader (m1n1) that runs a mini hypervisor with all kinds of bells and whistles for debugging and testing new kernels. This loads a modified U-Boot, but unlike an embedded device, the U-Boot stack provides a UEFI stub that allows mainstream Linux distros and their tooling to 'just work'. In August, the Asahi project announced that its reference distro was switching to Fedora. At this point, you can install the Fedora Remix without damaging an existing installation of Mac OS.

It seemed a daunting and very unrealistic effort, but the ringleaders are just brilliant. They're also creating a lot of great documentation. To try out Asahi's Fedora Remix, visit: **http://fedora-asahi-remix.org**."

Kernel Watch

Jon Masters keeps up with all the latest happenings in the Linux kernel, so you don't have to.

Linus Torvalds announced Linux 6.6-rc4, noting that the latest set of fixes were small and predicting larger sets of updates before the final 6.6 release. Thorsten Leemhuis's Regzbot regression tracking efforts say "there's not much to report wrt [with respect to] ... regressions introduced during the current cycle, as many of those I tracked were fixed". Fedora is on a quest to "confirm that the License field of all the RPMs is correct", and this extends to the kernel package. Prarit Bhargava let everyone know that "during our investigation we discovered two files in the kernel ... which from their descriptions imply that the code contained is not GPL". The offending files are firmwares for AppleTalk network devices, the drivers for which have not been updated since 2006. Given that AppleTalk is long since dead, the fast path option was to remove the drivers, but Jilayne Lovejoy also added links to a way that developers can request help adding correct licensing information in future situations.

> "Although DEC's venerable Alpha architecture is long since dead, it still has the most relaxed memory ordering and thus serves as the base for the formal Linux memory model."

Legacy architectures

A recent effort to clean up the kernel's Kconfig build system (which is also reused by a growing number of other open source projects) by reorganising some of the files caused the suggestion that all builds of Linux should by now support some variant of the PREEMPT kernel. Preemption is the concept of making the kernel able to interrupt even itself if a higher priority process needs time on the CPU, rather than waiting until the current activity has explicitly reached a suitable point to call into the scheduler. It leads to more responsive systems and thus is widely used.

There are different levels of preemption, up to and including an -RT (real-time) variant that also handles priority inheritance (bumping the priority of low-priority tasks so they don't block higher-priority ones, or cause priority inversions). Not all architectures were converted to add support for preemption, and this came up in discussion. Among the list was the venerable Alpha architecture from DEC (Digital Equipment Corporation). Although Alpha is long since dead, it still has the most relaxed memory ordering and thus serves as the base for the formal Linux memory model.

Matthew (Willy) Wilcox suggested the list "sounds like three-and-a-half architectures which could be queued up for removal right behind ia64". (We covered the removal of IA64 recently.) There is a point where architectures are so dead that their presence is entirely a maintainer burden but there was pushback here, such as, "The agreement to kill off ia64 wasn't an invitation to kill off other stuff," and "I find it unfair that maintainers of architectures [with] huge companies behind them use their manpower to urge less popular architectures for removal just because they don't have 150 people working on the port so they can keep up with design changes quickly." LXF

» ONGOING DEVELOPMENT

RISC-V patches include an updated (version 9) series adding support for future AIA (Advanced Interrupt Architecture) interrupt controllers, and SCS (Shadow Call Stack, similar to the x86 and Arm features). A number of posted patches implement support for custom vendor-defined ISA extensions, such as conditional arithmetic and conditional select/move. These are fairly fundamental in any architecture, so there is an implication of gratuitous (binary incompatible) differentiation. Indeed, the discussion covered this scenario, saying that vendors A and B might collaborate to add non-standard architecture extensions together. Which would be interesting.

Various updates to the Linux Plumbers Conference schedule were discussed, including the microconferences running alongside the main conference tracks.

Answers

Got a burning question about open source or the kernel? Whatever your level, email it to **answers@linuxformat.com**

Neil Bothwick is one of 30 clones whose hive mind solves your queries.

Q Fill yer boots

I cannot update the **linux-firmware** package on my Linux Mint 21 system. All other updates complete normally, but this errors out with the message 'No room left to install.' This makes no sense as I have a 1TB drive with plenty of free space. I can right-click and select Ignore The Current Update For This Package and everything appears to be running fine, but the next time, the update manager wants to update this again, with the same result.

How can I get rid of this error? If that is not possible, how can I tell the system to ignore this particular update in the hope that the next version will fix this error?
Cameron Knowles

A Since the introduction of UEFI booting, a separate FAT formatted partition at the start of the first drive has been required to hold the bootloader. Known as the ESP (EFI System Partition), this only needs to hold the bootloader, but as distros have traditionally set up a separate partition for **/boot**, the most common approach is to use this for **/boot** as well as the ESP (the bootloader files are located in **/boot/efi**). Very few packages touch this partition after initial installation, except the **linux-kernel** packages and **linux-firmware.** As **linux-firmware** is the only package reporting space issues, it is most likely that your **/boot** filesystem is full. Check this in a terminal by running:

```
$ df -h
```

This shows each filesystem's usage and it is likely that /, which occupies most of your 1TB drive, has plenty of space, while **/boot** is at least 99% full. How did this happen? It is because when the package manager installs an updated kernel, it does not remove the older one. This is a good thing – if there is a problem with the new kernel and your hardware, you may not be able to boot your computer, so you want the option of booting the previous kernel. All these kernels appear in the boot menu and you can see them in **/boot**, with names starting with **vmlinuz-** for the kernels and **initrd.img-** for the associated initial ramdisk files.

There are two ways to remove old kernels; in either case, make sure you are running the latest version. You can see which kernel you are running with:

```
$ uname -r
```

You should also check that your system is fully up to date. For a list of all installed kernels, run:

```
$ sudo dpkg --list | egrep 'linux-image|linux-headers'
```

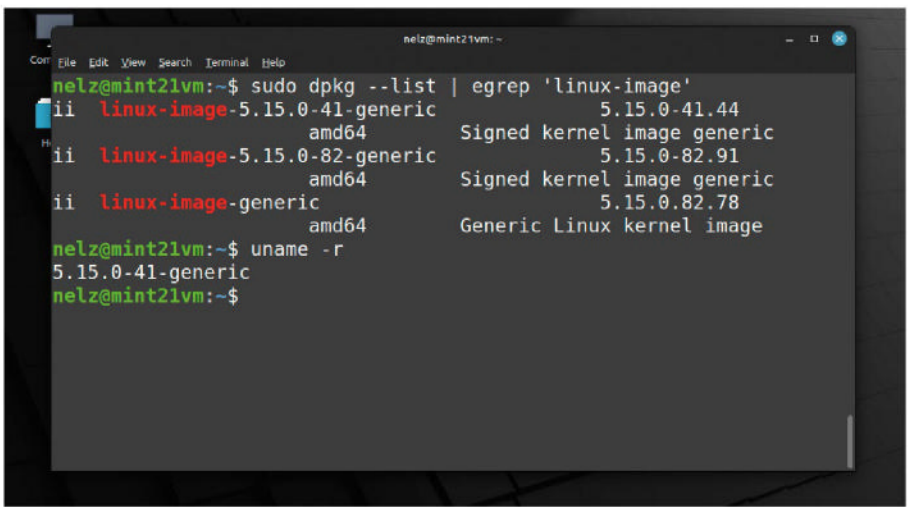
To remove all but the latest kernel, run:

```
$ sudo apt autoremove --purge
```

If you only want to remove specific kernels, using the package name as listed with *Dpkg*, remove them individually with:

```
sudo apt purge linux-image-x.x.x-generic
```

Do not be tempted to remove kernel or initrd files manually – this confuses the package manager as it keeps track of what it has installed.



Using Dpkg in a terminal shows which kernel packages you have installed. Keep too many and /boot can overflow.

Q Back and forth

I regularly write *LibreOffice* documents that contain a mix of English and Arabic. I have a keyboard shortcut set up within my distro (Elementary OS 5.1.7) so I can toggle between the keyboard layouts of these two languages by pressing both Shift keys together.

Within *LibreOffice* (version 7.5.5.2), I would like to assign the same keyboard shortcut to toggle the text direction. However, this particular combination doesn't appear to be available for assignment, and toggling between the text directions doesn't appear to be an option – you have to assign one keyboard shortcut for one direction, and another for the other direction.

Is there a way I can achieve this with my two-Shift-key shortcut?
James O'Hea

A It appears that what you want is not possible, for a number of reasons. Firstly, shortcuts in *LibreOffice* are bound to a combination of a modifier key and a character key, so using two modifier keys is not possible. Even it were, *LibreOffice* would then consume the keystrokes, so your desktop would not pick them up and change the keymap. We looked at whether it was possible to toggle the direction with a macro, but that would require the macro language to be able to determine the current setting in order to select the other, and that does not appear to be currently possible from our reading of the documentation.

However, there is a way to use a single shortcut to switch the keyboard layout and text direction to English, and another to switch the other way, by calling a script that uses *Setxkmap* and *Xdotool*. *Xdotool* manipulates windows on your desktop, including sending keystrokes to them. You

may need to install *Xdotool* first, then you can use a script something like this:

```
#!/bin/sh
setxkmap en
xdotool search LibreOffice key
Ctrl+Shift+a
```

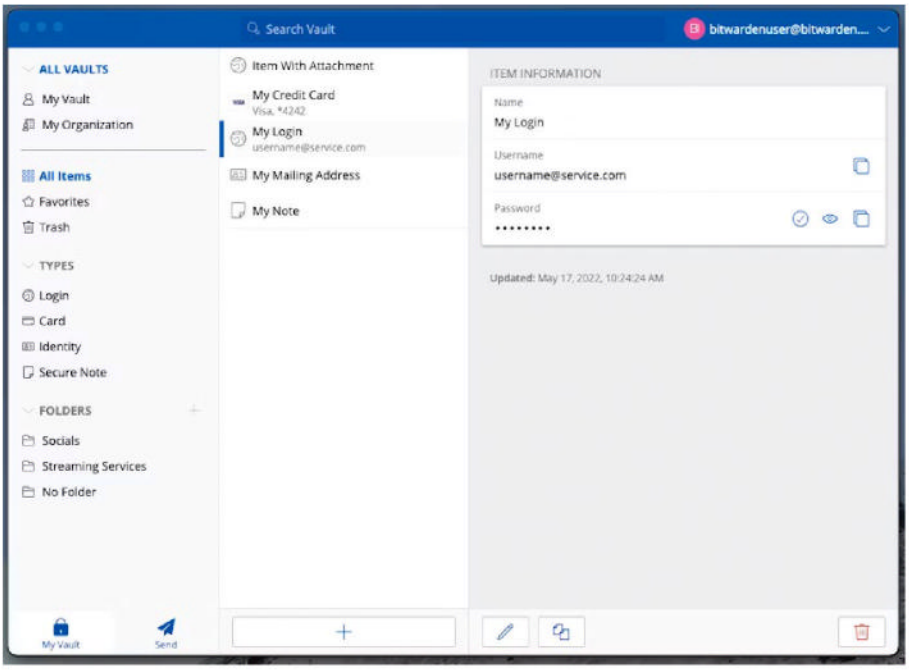
After setting the keymap to **en**, *Xdotool* first searches for a window with *LibreOffice* in its title, then sends the key combo given, which sets text input to left-to-right. You could make a second script to make the opposite change, or you could combine then into one script:

```
#!/bin/sh
case $1 in
 "l2r")
  setxkmap en
  xdotool search LibreOffice key
Ctrl+Shift+a
        ;;
 "r2l")
  setxkmap ar
  xdotool search LibreOffice key
Ctrl+Shift+d
  ;;
esac
```

Then you can call the script with either **l2r** or **r2l** as the argument to make the desired change. Use your desktop's settings to bind this to a hotkey. It means you need two hotkeys, but at least you know you are explicitly picking a particular setup; using one key combination to toggle could cause confusion if you get key bounces – or maybe it is only lousy typists like us who get too many of those.

Q Faulty memory problem

I have a Lenovo Thinkbook with Linux Mint Cinnamon installed. Somehow I have forgotten the login password that I set up, and now I can't get in. I was sure I'd written it down, but can't find it. I've read plenty of possible answers that include accessing the *GRUB* hub from the menu that opens when you self-install Linux Mint, but it doesn't work for this machine. I've also tried changing the password in BIOS, but it doesn't work for the login. *Kyle Lambert*



BitWarden is a free password manager that lets you keep track of the multitude of passwords we all need.

A The BIOS password has nothing to do with any of the passwords on any installed operating system; it controls access to the initial firmware menu, so do not lose it or you will have real problems. Because of the way passwords are stored on the system, retrieving a password from the system files is all but impossible to mere mortals. However, changing the password is simple. First you need to know which partition holds your root filesystem – you can see this with `df` in a terminal:

```
$ df -h /
```

Now get a live distro, it doesn't matter which one. Most distro install discs will do the job but our favourite for dealing with system repair issues is SystemRescue (**www.system-rescue.org**). Whatever you choose, boot the live distro and open a terminal. SystemRescue boots to a root console prompt by default; on graphical distros, you need to open a terminal and run the following to become root:

```
$ sudo -i
```

Now you need to mount your root partition, the one you identified before rebooting. For example, if it was **/dev/sda2**, you would run:

```
$ mkdir -p /mnt/mint
$ mount /dev/sda2 /mnt/mint
```

Running this should give similar figures to those you got before to confirm you've mounted the right filesystem:

```
$ df -h /mnt/mint
```

Now, using your username, change your password with the following:

```
$ passwd --root /mnt/mint username
```

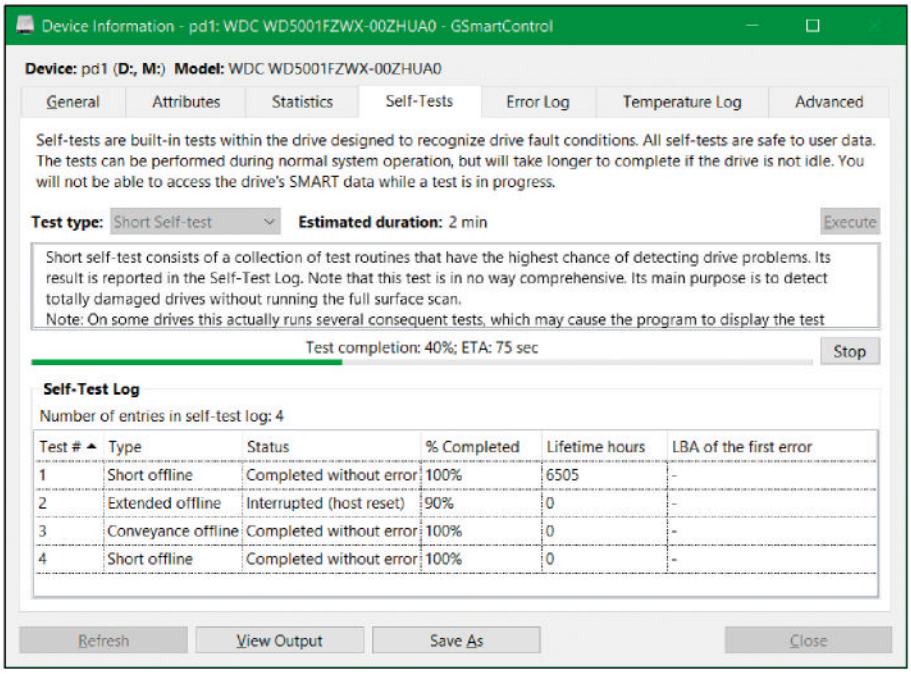
» A QUICK REFERENCE TO... TERMINAL HISTORY

The command line provides an efficient way to get things done, especially for repeated tasks, but it can demand a lot of typing, especially if you are a fellow alumnus of the two-fingered hunt-and-peck school of typing.

You have almost certainly discovered that you can use the cursor up key to revisit previous commands you've entered, which is fine for repeating what you did a few minutes ago, but the shell offers so much more. *Bash*, along with other shells, maintains a history file of the last 500 or so commands you typed, the exact number is determined by the **HISTFILESIZE** and **HISTSIZE** environment variables. We generally set these to 1,000, but you don't want to be hitting the up key hundreds of times to find the exact invocation of a command you used last week, so press Ctrl+R instead.

This starts the history search. Start typing part of a command and it shows the most recent match; as you type more, the match improves. If you want an earlier match, press Ctrl+R again. Once you have found the command you want, you can either press Enter to run it again, or the left or right cursor key to edit it first.

Some shell shortcuts use the history file, too, usually starting with `!`. `!-n` runs the nth from last command again, `!!` is a shortcut for `!-1` and runs the last command. `!xyz` runs the most recent command matching xyz, so is the same as Ctrl+R xyz Enter but suitable for scripting, too. There are lots more options to make your shell life easier, so check the *Bash* man page for full details.



GSmartControl provides a friendly face to Smartctl, so you can check your drives' health with a few mouse clicks.

You are now prompted for the new password. The `passwd` command normally operates on the running system, but the **--root** option tells it to modify the files of the system installed in the given directory. You do not give it the path to the actual password file (**/etc/passwd**) but the root directory of the system.

Reboot into Linux Mint and all should be well. We recommend using a password manager to store important information like this. *BitWarden* is an excellent, free, open source choice, or you can host your own server if feeling adventurous. Then you only have to remember one master password – but make it a good one!

Q Double vision

Running Debian testing, why am I logged in twice? If I enter `who` in a terminal it shows me on seat0 and tty2.

```
user@LinuxBox:~$ who
user  seat0  2023-08-12 01:05 (login screen)
```

Glenn Hastings

A The short answer is that you are logged in twice. The **seat0** entry is where you have logged in through the display manager, opening your desktop. Then you opened a terminal, which logged you into a separate session, the **tty2** entry.

While the terminal process is part of the desktop – log out of the desktop and the terminal closes – the session within it is not the same. To see this, run *Screen* in the terminal. Now `who` shows three logins for you. Log out of the desktop, without closing the *Screen* session, and log back in. Open your terminal and there are three logins again, even though your *Screen* session is not on the desktop. To get it back, run:

```
$ screen -r
```

This is because Linux is a multi-user system. Although that is usually thought of in terms of different users, it is also common to have multiple sessions for the same user. It should be noted that opening a terminal on a desktop doesn't always show as a separate login. For instance, when KDE Plasma opens terminal windows that were saved as part of the previous session, they are opened as part of the **seat0** process, but any opened from the desktop after that are separate.

Q Getting stressed

I am building a file and media server for my home network. I have bits and pieces of hardware left over from projects and upgrades that I would like to use, along with new components where necessary. I'd like to make sure everything works as it should before committing real data to the server. What is the best way to test the various bits of hardware?

Louie Lawson

A The main areas that cause problems are memory, CPU, disk drives and power. The last one is difficult to test and arguably the most critical – a poor power supply can cause all sorts of issues, so we recommend using a new, high-quality PSU. Memory is best tested with *MemTest86*. This is a bootable system that only uses the tiniest portion of your memory to load itself, leaving the rest free to be stress-tested. Many distro discs include it as a boot option, including our favourite rescue tool, SystemRescue (**www.system-rescue.org**). Boot from this and let it run for at least two full passes, overnight is best, as transient errors may not show up on a single pass.

There is a built-in diagnostic in modern hard drives called SMART (Self-Monitoring, Analysis and Reporting Technology). You need to enable SMART reporting in the BIOS, then you can test drives with *Smartctl*, which you may need to install from your distro's repos unless you are still working from something like SystemRescue. To test a drive, run:

```
$ smartctl --test=long /dev/sda
```

This runs the long test in the background and tells you how long it should take. You can see the results with:

```
$ smartctl --log=selftest /dev/sda
```

You can also get a quick health summary for a hard drive, after at least one test has been run, with:

```
$ smartctl --health /dev/sda
```

Get much more detailed information by replacing `--health` with `--all`.

For CPU testing, *Stress-ng* (**https://github.com/ColinIanKing/stress-ng**) is one of a number of choices. This command simultaneously runs several stress tests, in this case four CPUs, two for virtual memory and eight process forks, a good workout, and prints the results:

```
$ stress-ng --cpu 4 --vm 2 --hdd 1 --fork 8 --timeout 2m --metrics
```

It is a good idea to have *lm-sensors* running in another terminal to keep an eye on temperatures. Unlike the memory and disk tests, it is not always ideal to run a stress test like this unattended. LXF

GET HELP NOW!

We'd love to try to answer any questions you send to **answers@linuxformat.com**, no matter what the level. We've all been stuck before, so don't be shy. However, we're only human (although many suspect Neil is three kids in a coat), so it's important that you include as much information as you can. If something works on one distro but not another, then tell us. If you get an error message, please tell us the exact message and precisely what you did to invoke it.

If you have, or suspect, a hardware problem, let us know about the hardware. Consider installing *hardinfo* or *lshw*. These programs list the hardware on your machine, so send us their output. If you're unwilling, or unable, to install these, run the following commands in a root terminal and send us the **system.txt** file, too.

```
uname -a > system.txt
lspci >> system.txt
lspci -vv >> system.txt
```

Mailserver

WRITE TO US

Do you have a burning Linux-related issue that you want to discuss? Write to us at *Linux Format*, Future Publishing, Quay House, The Ambury, Bath, BA1 1UA or email **letters@linuxformat.com**.

Dull Chrome

I give up. I've been using computers since the late '60s, when I would program cards for the fledgling Midland Bank computers, then came programming games for my kids on a trusty Spectrum, and then the Acorn BBC, then moving on to Atari ST and early PCs, before 'graduating' to Unix systems and then finally Linux.

On my laptop I have Ubuntu 22.04 LTS and on my desktop I have Mint 21.03. I like both OSes, for different reasons. My default browser is *Google Chrome* because it synchronised all my devices and computers so that the same tabs, bookmarks, saved passwords and so on were all available regardless of what I am using – until now.

Suddenly, only on the Linux machines, *Chrome* has stopped working. It no longer displays images, around half of the bookmarks have disappeared, and most times buttons and the mouse pointer are not visible. And yet it continues to work perfectly on my (*whisper)* Windows 10 laptop, my Samsung tablets and phones.

So, my question is: what has happened to the Linux version of the *Chrome* install? All was fine until both the latest Ubuntu and Mint software updates, so I can only assume something was corrupted in the *Chrome* update. Am I the only one with this problem or is everyone else just using *Firefox, Opera* and so on?

Bob Biddles

Neil says...

I'll confess I use *Chrome* for office work because all the security systems are tied into the Google Account, but for personal and general browsing, I very much use *Firefox.* Web browsers are super-complex beasts and utilise their own renderer for graphics and the interface. It definitely sounds as though something has gone awry here. It could be that a driver or update to part of the display stack has tripped *Chrome* up.

Make sure everything is updated – your system and *Chrome.* Otherwise the easy option is to try toggling hardware acceleration. Select Menu > Settings > System > Use Hardware Acceleration. A more advanced option is to remove the cache:

```
$ cd ~/.config/google-chrome/Default
$ rm -rf GPUCache/
```

Then restart your machine afterwards.

Hardware acceleration has caused all sorts of Linux issues over the years.

Luking keys

I know a forceful shutdown means that memory can still be dumped, which can cause encryption keys to be compromised, but I haven't seen any information on whether either the kernel itself or other processes wipe things such as LUKS keys from memory before shutting down.

Securely wiping all of the memory before shutting down could cause slowdowns that are annoying and useless for 99% of users. Wiping LUKS keys should take a few milliseconds to seconds at worst, so I'm curious if that's already the standard or if even a gracefully shut-down computer would still be vulnerable to key-extraction via a cold boot.

Lenny Hartley

It turns out that LUKS keys should be gracefully wiped on power down.

Neil says...

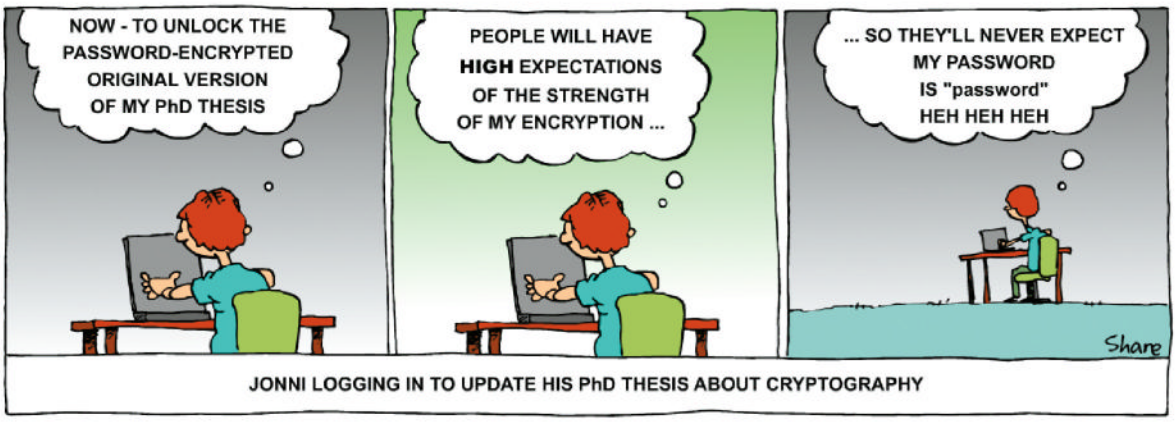
Encryption is hard – lots of very clever people with PhDs in cryptography (like our Jonni) have devised and attacked these systems over many decades now. So, generally, you can rest assured they're pretty secure on a day-to-day basis. Having said that, it's interesting that this very feature was added to initramfs (**https://github.com/dracutdevs/dracut**) back in August 2023, which is where it would have to be executed if it's to be included on encrypted root drives, too. So, it looks like your wish is coming true.

Linux confusion

I have been a Microsoft user for 20 years. During that time, I have built desktop computers, and I can usually troubleshoot most problems as well as repair them, so I consider myself quite competent with most things PC. Windows 10 annoyed me a lot, then Windows 11 put the last nail in the coffin. I have now turned to Ubuntu Linux Mint as the operating system most likely to satisfy my needs.

One thing that is right up there on my mind is the fact that I DO NOT WANT TO PLAY WITH PROGRAMMING! At 80 years old, I am just not interested, and you will appreciate that in Windows there is no programming involved. Therefore, if you, as

Helpdex



If we're saying to use the command line accounts for programming, Windows has that too...

the Linux community, want to win over more of us from the dark side, why oh why don't you make using Linux so much easier?

I try to read the magazine but I just get a headache with some 80% of the content, and rarely do I come across articles that will help translate my Microsoft mindset into a Linux one. There are also the many indecipherable acronyms that proliferate in your pages. You really are talking to us in a different language, and one that we cannot comprehend. Would it help if you placed more explanations of acronyms in the margins? Probably.

Programs that I love and have used for years are not available, or if an alternative is available – *GIMP*, for instance – this means another steep learning curve that I am reluctant to tackle. If I mention Wine and *Play on Wine*, this is yet another mystifying app that sometimes fails to work how you might expect. I honestly wish that Linux was easier.

A Confused of Ross-on-Wye

Neil says...

I don't think you have to program to use Linux, but there is a bit of that. Let's not overlook the huge advances the Linux desktop has enjoyed over the last quarter of a century. But even so, Linux is primarily a server OS – that's why it exists and why so many of its underlying tools are command-line based so everything can be scripted. I do sympathise, but let's not forget the Windows Registry and its own Powershell, if that's what we're calling programming.

I'll certainly try to put in explanations for the acronyms – in fact, I'd say we already do, though there's a limit to how much space we want to use on explaining what something is versus getting on with whatever we're trying to do. And I recommend learning *GIMP* – it's a bit grouchy and old (aren't we all?), but it does an excellent job. LXF

» LETTER OF THE MONTH

FOCAL point

Mike Bedford's article on FOCAL in **LXF301** reminded me of my senior year of high school in 1973. I took a course in Basic programming on a DEC PDP-8/e with 8k of magnetic core memory and two ASR-33 teletypes like the one pictured in the article. I remember there was a manual for FOCAL in the classroom and maybe a paper tape for it, but since it took 45 minutes to reload the Basic interpreter through the TTY's paper tape reader, I never saw it tried. Perhaps because it had the imprimatur of Dartmouth College, Basic was more popular with local schools. DEC computers were also quite common because they were (relatively) inexpensive and DEC seemed to promote them for educational use, perhaps hoping that students would favour them in their later careers.

After I started working, I bought a refurbished ASR-33 that I found on consignment in a local ham radio store. When I got into Linux, back when Slackware was loaded from a stack of floppies, I thought I might some day use the ASR-33 as a Linux terminal. I revived that thought when I started playing with Raspberry Pis. Then one night while I was trying to fall asleep, out of the blue it finally dawned on me: you may be able to run Basic or FOCAL on an ASR-33 teletype, but you can't run Linux through it!

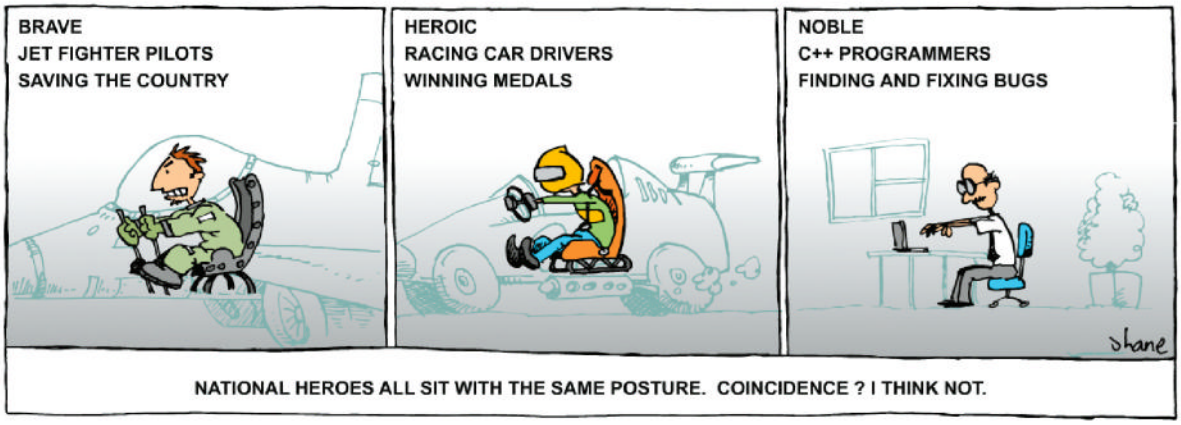
Geoff Gidman, Connecticut

Neil says...

It's really a delight that so many readers are getting a kick out of Mike Bedford's series on classic languages. If any of these have sparked memories or there's a machine or topic we've not covered, please let us know!

Here's a stack of punchcards ready for the DEC PDP-8. One is a FOCAL interpreter.

CREDIT: Wolfgang Stief, CC BY 2.0, www.flickr.com/photos/stiefkind/16756514468/



shane_collinge@yahoo.com

» PLUS: Exclusive access[1] to the *Linux Format* subs area!

Thousands of DRM-free PDF back issues and articles! Get **instant access** back to issue 66 (May 2005) with tutorials, interviews, features and reviews. At **linuxformat.com**

1) Only available to www.magazinesdirect.com subscribers

» CHOOSE YOUR PACKAGE!

One payment. A whole year of issues delivered straight to your door or device. Simple!

ANNUAL PRINT EDITION

Only £62.99

13 issues of *Linux Format* in print over 12 months, **saving 25%**!

ANNUAL DIGITAL EDITION

OUTSIDE THE UK? Turn to **page 65** for more great subscriber deals!

Only £51.99

13 issues of digital *Linux Format* over 12 months, **saving 42%**!

*Terms and conditions: **Offer closes 31/12/2023**. Offer open to new UK subscribers only. Pricing is guaranteed for the first 12 months and we will notify you in advance of any price changes. Please allow up to six weeks for delivery of your first subscription issue (up to eight weeks overseas). Your gift will be delivered separately within 60 days after your first payment has cleared. Gifts only available to subscribers on the UK mainland. Gift not available with a digital subscription. Gifts only available in the colour black. The full subscription rate is for 12 months (13 issues) and includes postage and packaging. If the magazine ordered changes frequency per annum, we will honour the number of issues paid for, not the term of the subscription. For full terms and conditions, visit www.magazinesdirect.com/terms. For enquiries please call: +44 (0) 330 333 1113. Lines are open Monday-Friday 9am-5pm UK time or email help@magazinesdirect.com. Calls to 0330 numbers will be charged at no more than a national landline call and may be included in your phone provider's call bundle.

Winter Savings

SUBSCRIBE FROM JUST £6

PLUS OFFERS ON BINDERS, GUIDES & SPECIALS

See the entire range online at

MAGAZINESDIRECT.COM/WINTER23

or phone **0330 333 1113**

Outside of the UK? See our international offers online

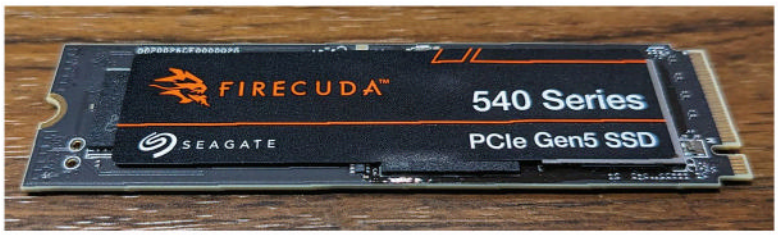
Seagate FireCuda 540

Chris Szewczyk finds high performance and peace of mind. At a price.

SPECS

Size: 2TB (1TB option)
Type: PCIe 5.0, NVMe 2.0
IC: Phison PS5026-E26
Mem: Micron 232-layer 3D TLC
Seq: 10,000MB/s read/write
Endurance: 2,000 TB TBW

PCIe 5.0 SSDs have been teased for what seems like an eternity. In spite of all the chat, few drives are available, even less so from the big-name SSD makers. Seagate belongs in that category, and its FireCuda 540 leads the Seagate charge into the brave new world of PCIe 5.0 SSDs. For now, that new world is a very expensive one to buy into. All PCIe 5.0 SSDs carry a massive price premium over PCIe 4.0 drives.



Two terabytes of data have never copied so quickly.

Seagate is positioning the FireCuda 540 as an all-round option. There are obvious benefits but it also carries an excellent endurance rating and comes with three years of data recovery support. That kind of support and peace of mind will appeal to professionals, and they're less likely to worry about price if you think of Seagate's impressive warranty as a service.

This review unit doesn't include a heatsink so it's imperative the motherboard has an integrated cooling solution. If not, you risk throttling or possibly long-term damage if you don't have good case airflow. Even if your mobo has M.2 cooling support, never install any high-performance SSD directly below a graphics card.

Performance art

The FireCuda 540 includes the latest and greatest 232-layer Micron TLC NAND, plus 4GB of LPDDR4 memory. Seagate opted not to pursue maximum performance at all costs in favour of a blend of reliability and performance, a point in the drive's favour.

The performance of the FireCuda 540 is a mixed bag. There's a reason vendors like to push sequential performance as the most important measure of an SSD's performance. The FireCuda does really well here. The superior controller, 232-layer NAND, DRAM and cache all give the FireCuda 540 a huge advantage over previous-generation drives in sequential performance and internal file transfer speed. Simultaneously reading and writing 30GB made up of over 25,000 files in 50 seconds is impressive.

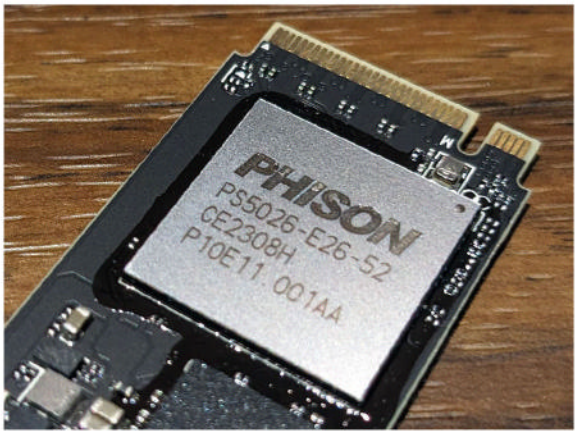
But sequential performance is only part of the story. The FireCuda still outperforms PCIe 4.0 drives in random tasks, but not by anywhere near as much as you'd hope, given the price. And you won't notice any real difference in the perception of responsiveness.

Game load times look good for the FireCuda 540 but it's up to you to decide whether a second or two here or there is worth the premium price being asked for the FireCuda. We'd say it isn't. Not yet, anyway.

It's been said many times that PCIe 5.0 drives run hot. Our ASRock X670E Taichi motherboard comes with a fairly standard integrated heatsink, but we took the liberty of testing the drive with ASRock's Blazing M.2 heatsink, which includes a small embedded fan. That took 13°C off of the peak temperature, but it had little effect on the performance of the drive, probably because the standard Taichi heatsink does just enough to prevent it from throttling, reaching 80°C.

At well over twice the price of the best PCIe 4.0 drives, PCIe 5.0 SSDs are a tough sell. Seagate is not alone here. They're all too expensive. A look at the few PCIe 5.0 drives that are available puts them all around the same price. So, in relative terms, at £287 the 2TB FireCuda 540 isn't unreasonably priced.

In time, the price of the FireCuda 540 will fall. And it really needs to if we're to recommend it. For now, stick with a high-performance PCIe 4.0 drive like Seagate's own FireCuda 530, which is one of our favourites. LXF



Helping drive those bits around at speed.

VERDICT

DEVELOPER: Seagate
WEB: www.seagate.com
PRICE: £287

FEATURES	9/10	EASE OF USE	9/10
PERFORMANCE	9/10	VALUE	6/10

Given the huge prices of PCIe 5.0 drives, it doesn't make a lot of real-world sense, but if you want an ultra-fast SSD with an excellent endurance rating and warranty, why not?

» Rating 7/10

AMD Radeon RX 7800 XT

Introducing a multi-chip RX 6800 XT for less – and **Dave James** thinks that it's actually the best £500 GPU around.

SPECS

GPU: Navi 32
Arch: RDNA 3
Process: TSMC 5nm + 6nm
Clock: 2,124MHz (2,430MHz boost)
Mem: 16GB GDDR6, 256-bit, 624GB/s
Stream: 3,840
Compute: 60
RT cores: 60
AI cores: 120
ROPs: 96
Cache: 64MB
TGP: 263W, 2x eight-pin
Die: 200 + 150mm²
Trans: 28.1B
Codec: 4K h264, h265/HEVC, AV1

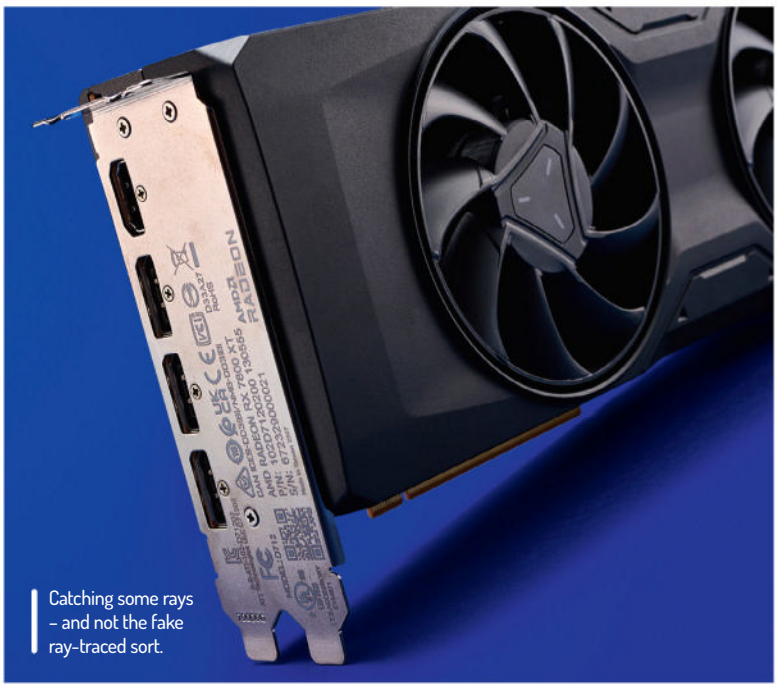
How much value do you put on ray tracing as a gamer? That's got to be the question at the forefront of your mind if you're considering dropping £500 on an AMD Radeon RX 7800 XT. Because if you are 100% sold on ray tracing, you probably need to consider whether you're actually willing to pay another £100 for the privilege of extra RT performance and drop the cash on the RTX 4070 instead.

If, however, you're convinced rasterised performance is the only GPU metric worth a damn in this topsy-turvy world of gaming, then the RX 7800 XT is probably the best upper-mid-range graphics card you can buy today.

Now, apologies for sounding too much like some class-obsessed Victorian English gent, creating strata after strata of card so we can jam the RX 7800 XT somewhere that makes sense. But when graphics card pricing is so ludicrously spread out – with the next Radeon up costing £800 at best – this £500 price point needs some sort of classification of its own.

It's not high-end any more, and it's not really enthusiast, because that denotes some level of decent 4K performance. And for this amount of cash, we're certainly not calling it mainstream. Although it is essentially now the middle of the Radeon RDNA 3 line-up, calling the RX 7800 XT mid-range doesn't sit well with us, either. So, we can go for lower-high-end or upper-mid-range, and we think we'll stick with the latter.

The Radeon 7800 XT outside of its natural environment, the PC case.



Catching some rays – and not the fake ray-traced sort.

Gentleman's relish

So, that's where the RX 7800 XT lives, launching with a reference price of £479 for the modern English gent and with the sort of gaming performance that means it can mix it with some of the best GPUs of this and the past generation. Effectively, now we have four graphics cards capable of very similar frame rates: the RX 7800 XT here, the RTX 4070, and then the RTX 3080 (which is pretty much end-of-life now) and RX 6800 XT from the previous generation.

The reductionist way of looking at the RX 7800 XT is that this new generation of Radeon GPU is delivering the sort of performance we had to pay £650 for last time around. But that's still a pretty tantalising way of looking at things. By offering frame rates akin to both the RX 6800 XT and RTX 3080, this price point is looking more potent than ever before.

And in terms of current-gen head-to-head with the RTX 4070 – itself delivering RTX 3080 frame rates with benefits – you're getting straight raster performance in games that is broadly similar (and sometimes higher) but for £100 less.

That should all make it a very straightforward win for AMD, and the RX 7800 XT is a card that ought to fly off the shelves. Were it just about the raw silicon, maybe that would be the end of it, but there is more to the modern GPU ecosystem than just stream processors these days, and it would be foolish not to talk about the competing feature sets and strengths of AMD and Nvidia's latest cards.

However much AMD would like you to know that it has improved the ray-tracing performance of its RDNA 3 architecture, with silicon dedicated to accelerating the RT pretties, it's still a generation behind Nvidia, at best –

when you throw heavily ray- raced games into the benchmarking mix, Nvidia's RTX 4070, and to a lesser extent the RTX 3080, takes the lead. But as the vast majority of games don't utilise ray tracing, never mind Linux-supporting games, this puts far more positives in the RX 7800 XT corner than the competing RTX 4070. It regularly wins out on straight raster frame rates, it has more video memory, which is a consistent theme of gaming today, and it has its own solid upscaling tech in FSR 2.2, with the v3 update shortly on the way. And it's £100 cheaper, which is probably the real kicker.

What AMD wants you to know is that this is a 1440p graphics card, designed to top 60fps in the latest titles at the top settings. And it goes well beyond that outside of that brutal *Cyberpunk 2077* outlier. The top four cards we tested – the RTX 3080, RTX 4070, RX 6800 XT and RX 7800 XT – all trade blows throughout the benchmarks at this level and that just shows how close these cards are, and it's pleasing to see the RX 7800 XT as the cheapest one sometimes leading the way, too.

It's also important to note just how far ahead of the next one down the GeForce stack this Radeon GPU is. Compared with the RTX 4060 Ti, whether 8GB or 16GB, the RX 7800 XT is around 25% faster for another 25% on the sticker price. And its relative performance against the RX 7700 XT is a nail in the coffin for its Navi 32-toting brethren. On average we saw a performance deficit of around 20%, but the price delta is a mere 10%.

That makes the RX 7700 XT's ludicrous £449 launch price too close to the RX 7800 XT for it to be relevant. Here's hoping it follows the RX 6700 XT's trajectory and drops to around £350, otherwise it really is a DOA GPU.

Raw power

And all these numbers represent the RX 7800 XT's native performance, so are all just down to what the raw silicon is capable of without resorting to upscaling shenanigans for extra performance. Of course, you absolutely can do that with the RX 7800 XT, though it's less of a given that you'll enable FSR with AMD's implementation not faring too favourably compared with Nvidia's DLSS. That's especially true considering a lot of FSR-supporting games have still not been updated from the shaky first-gen version of the Radeon tech.

The RX 7800 XT even has some 4K gaming chops, as you would expect from a card that's offering performance right up there with the high-end of AMD's previous generation of GPUs. At this level, you are going to need some upscaling to deal with any ray-traced lighting effects, however, because away from pure rasterised rendering, you can see the RX 7800 XT start to chug with this many pixels on screen.

But, at this level, upscalers have a lot more image data to work with, so your 4K upscaled experience is going to look a whole lot better than a 1080p upscaled one. We mean, that's just gross, especially on FSR 1.0.

On the more holistic side of things – temps and power draw – the advanced Nvidia Ada architecture is the clear winner. Its RTX 4070 is able to deliver the same general experience, with the extra ray-tracing cherry on top, and still come in far less thirsty and far cooler as well. If we're talking about a better GPU architecture, Ada is probably that. Though we guess that's why Nvidia has opted to price it so high. For anyone with a budget of £500 to spend, the RX 7800 XT is the only card on the block. LXF

» UNDER THE SHELL

Here we have the Navi 32 GCD, which is two thirds the size of that inside the Navi 31 chip at 200mm^2 versus 300mm^2. Around that core chip are four Memory Cache Dies (MCD), which house the Infinity Cache, VRAM and the memory bus.

The GCD houses a total of 60 compute units (CUs), and therefore 3,840 shaders, or stream processors; that's the same as the RX 6800 (non XT) from the previous generation. But the base architecture of RDNA 3, those CUs, have been enhanced to the tune of "approximately 17.4% architectural improvement clock-for-clock". So sayeth AMD, which sounds rather specific for an approximate measure.

And in terms of those clocks, the RX 7800 XT runs at a nominally higher frequency than either the RX 6800 or RX 6800 XT it's replacing. Though, notably, not as high as AMD initially promised for RDNA 3. The generational progress, therefore, feels rather slight by direct comparison.

You are still getting 16GB of GDDR6 via those four MCDs, and also an aggregated 256-bit memory bus alongside them. That's pretty damned good for a £500 card, and it's a proper 16GB, too, not clam-shelled in some weird design à la the now-£449 RTX 4060 Ti and its meagre 128-bit bus. It's also higher clocked memory than the RDNA 2 cards, running at 19.5Gb/s against the 16Gb/s of the RX 6800-series. That all helps bump the effective memory bandwidth, too, along with the second-generation Infinity Cache, of which there is surprisingly less.

Expect a deluge of third-party 7800 XTs with fancy lighting and the like.

VERDICT

DEVELOPER: AMD
WEB: www.amd.com
PRICE: £479

FEATURES	**9/10**	**EASE OF USE**	**9/10**
PERFORMANCE	**8/10**	**VALUE**	**8/10**

A very good graphics card for non ray-traced gaming and the highest performance from a sub-£500 card.

» Rating 8/10

Qubes 4.1.2

Nate Drake explores the apex of Linux security: an OS that divides all your apps into colour-coded, isolated virtual machines.

IN BRIEF

Qubes does something we thought impossible. It provides next-level security with easy management from a centralised colour-coded interface. A hefty ISO and lack of multi-boot are a small price to pay.

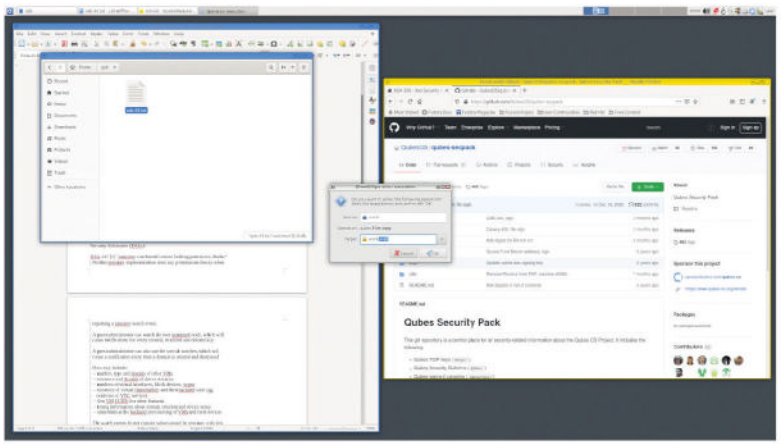
SPECS

CPU: 2GHz (Intel VT-d/ AMD IOMMU)
Mem: 6GB
HDD: 32GB
Builds: x86_64

While there are numerous security-focused Linux distros, Qubes can be compared to Fort Knox. This Fedora-based OS was first produced in 2012. The latest testing release (4.2.0-rc3) was made available in September, and will go through at least one more round of tests before a stable version is made available. We've focused this review on the latest stable version of Qubes (4.1.2), which was released in March.

Qubes uses so-called security by isolation to prevent apps from interacting in a way that could compromise your privacy. The OS achieves this through using what it calls qubes, implemented as lightweight Xen virtual machines.

These qubes can be categorised based on levels of trust. For instance, you can have one for work activities, one for home, one for shopping and so on. The qubes VMs can be based on Fedora 37, like Qubes itself, or other popular OSes, such as Debian 11 or Windows. The OS can also create other service qubes for specific purposes, such as managing the network stack.



Qubes VMs are based on predefined templates. By default, these are Whonix 16, Fedora 37 and Debian 11. Windows is also supported.

Multiple choice

The advantage of this is clear: you can divide your device into multiple machines separating different aspects of your digital life. This is even true when using the same executable. The *Firefox* browser in one domain, for instance, has no access to files used by another version of *Firefox* running in another domain.

This may sound convoluted but to quote the Qubes website, "All windows are displayed in a unified desktop environment with unforgeable colored window borders so that different security levels are easily identifiable."

While we're on the subject, the Qubes website provides an excellent overview of what the OS does, as well as a great Getting Started section for newcomers. We visited this page to learn about app qubes. These are based on a predefined template and intended for running programs such as browsers and email. If one of these app qubes is compromised, it doesn't affect the others. You can create them with the Template Configuration wizard that appears at first boot.

Qubes does, however, have a single admin qube, dom0. From here you can administer the entire system. So, if dom0 is compromised, the whole system is unsafe, but the OS is designed in such a way to prevent this – for instance, dom0 has no network connectivity and is used only for running the desktop environment and window manager. The default environment is Xfce (KDE is also supported). Qubes supports both the *i3* and *AwesomeWM* window managers.

You can use the admin qube to select the frame colour of each of your qubes, so you can tell them apart. For instance, most users choose red to indicate qubes that run dangerous or untrusted applications.

From the main desktop, you can access a number of Qubes-specific tray widgets. These include Whonix SDWDate, which allows you to manage tor connections for the Whonix qube. Qubes Clipboard lets you easily copy text from dom0, while Qubes Devices enables you to attach or detach devices to specific qubes.

The Qubes Domains widget enables you to manage running qubes, turn them on and off, and monitor RAM and CPU usage. You can also gain an overview of this through Qubes Manager, which shows the status of all qubes, even if they're not currently running. You can also add and remove qubes from here.

It's clear that the Qubes developers take security seriously and have done everything they can to make juggling multiple, isolated VMs as painless as possible. Whether you need this level of security is up to you. LXF

VERDICT

DEVELOPER: The Qubes OS Project
WEB: https://qubes-os.org
LICENCE: GPL v2

FEATURES	9/10	EASE OF USE	7/10
PERFORMANCE	8/10	DOCUMENTATION	10/10

Qubes compartmentalises your digital life into isolated virtual machines. It may be overkill but it's easy to use.

» Rating 9/10

Mageia 9

Nate Drake visits the spiritual successor to Mandriva Linux and comes away enchanted.

IN BRIEF

Mageia is derived from the Greek word for magic and it's easy to see why: installation media for every taste, plus a revamped setup wizard. The Control Center also makes configuration easy.

SPECS

CPU: 1GHz
Mem: 512GB (2GB recommended)
HDD: 20GB
Builds: i586, x86_64

In 2010, the company that owned the assets to the popular and easy-to-use Linux distro Mandriva was liquidated. Shortly afterwards, a number of former Mandriva employees and volunteers from the Linux community announced the creation of Mageia. Unlike Mandriva, this OS is backed by a non-profit organisation based in France (Mageia.org). The project snowballed and to date hundreds of developers have collaborated to produce nine stable versions of the OS every one to two years.

The new welcome wizard walks users through the Mageia operating system and explains various features, such as the Control Center.

If you decide to take the latest Mageia 9 for a test spin, there are a number of installation media to choose from. Classic ISOs weigh in at just over 4GB and can be used to install the OS directly. Impressively, there are versions available for 167 locales. There's also a minimal network install ISO of around 50MB, plus live ISOs to try without installing. The latter are available with Plasma, Gnome or Xfce desktops.

On first boot of the Plasma live ISO, we were very impressed to see a colourful welcome wizard that explains how live mode works, as well as pointing to Mageia's extensive online documentation for help with installation. This has been revamped since the previous Mageia release to provide a more linear progression through the OS's features. For instance, the wizard has a section on the MCC – the Mageia Control Center. This is a set of tools that handles system-wide settings, such as software, hardware and network management. In the latest version of Mageia, the sound manager, *Draksound,* has been upgraded to allow users to switch easily between PulseAudio and PipeWire.

Package management is handled by *RMPDrake,* although the setup wizard also supports installing *Dnfdragora* if you prefer. The Mageia 9 release notes that 32-bit repositories are disabled by default but you can enable them again if apps like *Steam* require it. There's also a link to the Mageia wiki, listing key available apps. Under the hood, RPM DB uses SQLite, instead of the "old and unmaintained" Berkeley DB.

Part of the reason we chose the Plasma version of Mageia is that the NetworkManager system service is now enabled by default on the live ISO. Mageia 9 uses Plasma 5.27.5, built on top of *Qt 5.15.7.* The default display manager is SDDM (Simple Desktop Display Manager) but you can install Wayland if you wish.

LibreOffice has been updated to version 7.5 and now even includes a voice dictation tool. *Chromium* has been retired in favour of the latest ESR (Extended Support Release) of *Firefox. Vim* has also been updated to version 9.0. Media playback is handled by the ever-redoubtable *VLC Media Player (v3.0.18 'Vetinari').* Our KDE-based version of Mageia also bundled *Dragon Player* and *Elisa* (both v23.04.1).

System settings are also extensive, though we note there was only one alternative desktop wallpaper available with both light and dark themes. Game lovers will be disappointed to find that none seem to come bundled with Mageia 9, but the wiki explains many are available for download in the Games category of the package manager, which we could view after manually selecting our nearest media mirror via the MCC.

Mageia also appears to take network security seriously. The MCC enables you to set up your very own firewall *(IPV6* firewall is also available). Users can set authentication to access Mageia tools and there are even parental controls. At the time of writing, the Mageia install doesn't support automatic FDE (full disk encryption), but the release notes mention encrypted installs, suggesting it can be done manually. LXF

VERDICT

DEVELOPER: Mageia.org
WEB: www.mageia.org
LICENCE: GPL

FEATURES	8/10	EASE OF USE	10/10
PERFORMANCE	8/10	DOCUMENTATION	8/10

Mageia truly is magic. There are numerous flavours and the new setup wizard walks new users through various features.

» Rating 9/10

Ubuntu 23.10

Nate Drake dives into the labyrinth and finds Canonical has released another solid offering with its Mantic Minotaur iteration of Ubuntu.

IN BRIEF

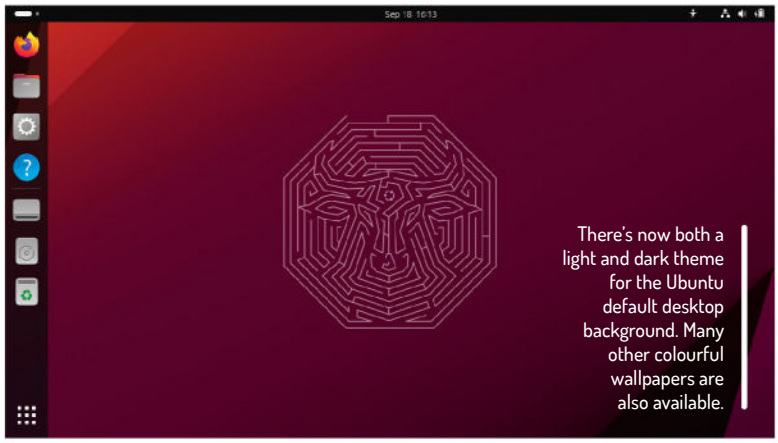
Far from being labyrinthine, Ubuntu 23.10 offers an easy setup, along with a revamped Gnome desktop, making tiling and arranging workspaces much easier. Searches are global and there's a new App Center.

SPECS

CPU: 2GHz dual-core
Mem: 4GB
HDD: 25GB
Builds: x86_64, ARMv8, AArch64

The Ubuntu blog describes the word Mantic as being "used to describe one's ability to prophesy or reach into the future". This is exactly what the **LXF** team is attempting to do in test-piloting a daily build of the OS prior to the official release on 12/10/23.

The Minotaur was a mythical half-man, half-bull who guarded the halls of the impenetrable labyrinth of Crete. Luckily, this is totally at odds with our experience of Ubuntu 23.10 Mantic Minotaur, which fully lives up to the OS's philosophy of openness and ease of use.



There's now both a light and dark theme for the Ubuntu default desktop background. Many other colourful wallpapers are also available.

We're partial to new desktop backgrounds and found the default understated image inspiring. There's also a less distracting dark variant. If you prefer something less abstract, there are 10 other Minotaur-themed wallpapers, including one rendered lovingly in pixel art.

While we're talking about desktops, the version we chose for review uses Ubuntu's default Gnome environment, updated to version 45. Changes include the new Tiling Assistant extension. This means that by default Gnome tries to solve the issue of floating windows by adding a Windows-like snap assist to the desktop, enabling you to drag windows to automatically use quarter or half-tiling on your workspaces. Tiling pop-ups and tiling group features are also supported, enabled or disabled by going to Settings > Ubuntu Desktop.

The Activities overview button has been replaced with a sleek workspace indicator. Your current workspace appears as a small white bar, while others are represented by a grey dot. This makes it much easier to tell which workspace you're currently using.

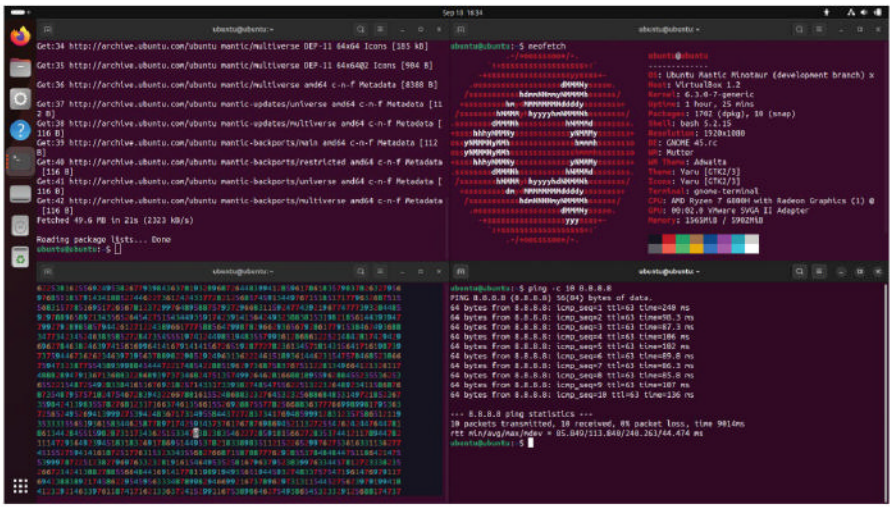
Opening the *Nautilus* file manager also reveals some of the most important updates. This includes a more stylish look and feel, and you can now also run a quick global search of all files and folders, instead of searches being restricted to the **Home** folder, as before.

Although it wasn't present in our daily build of Ubuntu 23.10, *Nautilus* on Gnome 45 will also benefit from a sidebar that extends to the full length of the left-hand side of the window, now renamed Places. The three-dot menu at the top-right of the windows will also be replaced with a hamburger menu in the left-hand pane.

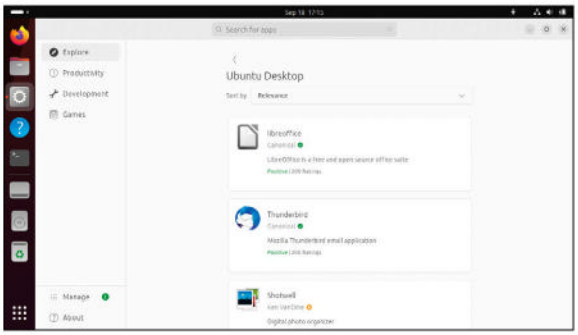
Since we're talking panes, be sure also to check out Gnome Settings, which uses a better contrast to distinguish between the left pane and the main window.

Ubuntu 23.10 also boasts a new App Center built using *Flutter*. This makes it easier to navigate programs. Categories include Explore, Productivity, Development and Games. The Manage section enables you to check for and apply Snap updates. As things stand, this seems to live up to Ubuntu devs' earlier statements that it would be a "snap-first" store. (Flatpak was honourably retired in the previous version of Ubuntu and developers have specifically confirmed it won't be reintroduced to the App Center.) Debian package lovers, however, might not be disappointed, because there are rumours that DEB support will be included in the final stable release of Mantic. Our only real criticism is that the App Center seems to lack its own dedicated icon.

Mantic uses Gnome 45, which now includes the tiling assistant for easy half and quarter-tiling of windows.



The App Center contains an Explore section with a dedicated Ubuntu Desktop area. This is very welcome, as Canonical has actually removed a number of bundled apps for Ubuntu 23.10. Chief among these is the *LibreOffice* suite, but others haven't escaped the chop, including webcam app *Cheese*, the *Transmission* BitTorrent client and *Document Scanner*. Other casualties include the *Rhythmbox* music player, the *Shotwell* image organiser and preinstalled games like *Mines* and *Sudoku*. Of course, these can all be downloaded again, but given the Mantic



The new App Center is a less buggy alternative to the previous Software app. Use it to install missing core apps.

ISO weighs in at 4.4GB at the time of writing, we aren't sure what the benefit of removing apps is to end users.

Beneath the hood, Ubuntu 23.10's *Mutter* window manager introduces YUV colour support in Wayland. This allows apps to offload YUV-to-RGBA conversions to the compositor, which is the format used by most modern displays. In theory, this can make for improved system performance and a lower energy footprint.

Crucially, Mantic Minotaur also has a new version of software properties, which hugely changes the way the OS manages PPAs (Personal Package Archives).

Traditionally, Ubuntu PPAs have been managed via a LIST file located at **/etc/apt/sources.list.d/**, accompanied by a *GPG* keyring stored in **/etc/apt/trusted.gpg.d**. As of Ubuntu 23.10, PPAs are added as deb822-formatted . sources files, where the keys are directly embedded into the file's Signed-By field. Benefits of this include the fact that removing a repository now always removes its associated key. Keys are also unique to each PPA, so can't be used to sign other repos.

Mantic also now ships with Python 3.11 by default, saving you the trouble of installing it manually, as in the past. The distribution also incorporates PHP 8.2, GCC 13 and LLVM 17.

As any good Linux user knows, Ubuntu is available in any number of flavours. Stable distros using alternative desktop environments will also be made available for download on 12th October.

One of the more significant builds is Ubuntu Budgie, which now comes with version 10.8 of the desktop environment. It now supports Magpie, has improved menus and system tray, and even an upgraded trash applet that enables you to recover deleted files. Xubuntu 23.10 will use version 4.18 of the Xfce desktop, which includes more colourful wallpapers, multiple image preview panes and customisable toolbars in *Thunar,* and provisional Wayland support. Lubuntu 23.10 uses the LXQt 1.2.0 desktop and also supports Wayland, and includes greater customisation of search history and app settings. Meanwhile, Kubuntu will ship with version 5.27.5 of the KDE Plasma Desktop, which includes its own window tiling system, more stylish app themes and better support for Flatpak permissions. Whichever you choose, Ubuntu 23.10 will be supported for nine months because it's not an LTS (Long Term Support) release.

Overall, we came away from Mantic Minotaur impressed. True, setup is now a little more involved as certain core apps have been removed. Still, this is only an issue for those doing a clean install. The App Center also makes it very easy to navigate and install programs, and we hope to see both Snap and DEB support in the final release. Gnome 45's new bundled tiling features will also make for much more productive workspaces. LXF

» TPM-BACKED ENCRYPTION

On 7th September, Canonical announced that TPM (Trusted Platform Module) full-disk encryption would be coming to Ubuntu. Traditionally, full-disk encryption has relied on entering a passphrase at boot time using LUKS, meaning protection is really only as good as the passphrase length and complexity.

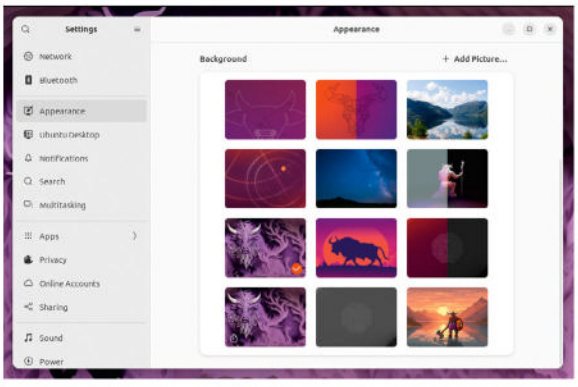
Devices that incorporate a TPM chip, however, can create cryptographic keys and encrypt them, so that the keys can only be decrypted using a special 'root key'. This can be used for operations such as securing and decrypting a hard drive.

Ubuntu Core's FDE support already requires both UEFI Secure Boot and TPM 2.0. As of Ubuntu 23.10, Canonical has stated that it also wants to bring TPM-backed FDE to the Ubuntu Desktop.

This provides much better protection against so-called evil maid attacks, whereby someone with physical access to the machine manipulates it in some way– for example, by installing a bootkit. TPM-backed FDE supports authenticating initrd and other components using a series of cryptographic hashes and signatures to provide a verified boot process. Another huge advantage is that users no longer have to enter a separate passphrase at boot time.

Canonical stresses that this feature is still experimental, so we recommend you don't use it to secure any sensitive data until there's a stable release. If you opt to install TPM-backed FDE, the kernel and bootloader assets are loaded from Snaps instead of DEBs. Alternatively you can opt for the default non-TPM version of Ubuntu, which is an entirely DEB-based classic desktop system.

Read more at: **https://bit.ly/lxf308tpm**.



Settings has also received an overhaul. Click Appearance to change the background or Ubuntu Desktop to configure tiling.

VERDICT

DEVELOPER: Canonical
WEB: https://ubuntu.com
LICENCE: Various GPL/BSD

FEATURES	**9/10**	**EASE OF USE**	**10/10**
PERFORMANCE	**10/10**	**DOCUMENTATION**	**10/10**

Mantic's new, bug-free App Center makes installing new programs a breeze. The Gnome 45 desktop is also packed with useful features.

» Rating 10/10

Old World

A 4X (Explore, Expand, Exploit and Exterminate) game? Management is all ears and demands **Fraser Brown** plays it.

SPECS

Minimum
OS: Ubuntu 18.04+ 64-bit
CPU: Intel Core i5 6400
Mem: 8GB
HDD: 10GB
GPU: Nvidia GeForce 770, AMD Radeon R9 290, Vulkan required

Recommended
OS: Ubuntu 20.04+ 64-bit
CPU: Intel Core i5 8500
Mem: 16GB
HDD: 14GB
GPU: Nvidia GeForce 1060, AMD Radeon RX 5500

The story of our first campaign in *Old World,* where we led our Greek civilisation to global domination, is equally the story of Rome, our greatest nemesis in this sprawling turn-based 4X. When we first met the Romans, they were extremely friendly, offering gifts and hospitality, but it was a poisoned chalice – literally. A sickness spread, and Roman gestures of friendship were the source. We demanded justice and compensation; Rome wanted war. It would take nearly 200 years until we got our revenge.

War, what is it good for? Well, crushing your enemies, seeing them driven before you, and hearing the lamentations of their women, obviously.

Despite using the language of *Civilization* – from workers to wonders – it should be starting to become clear that Soren Johnson and Mohawk Games' take on the historical 4X formula is different from Firaxis's approach. We still build cities, conquer other ones, develop the tiles around them, and along the way determine the cultural and scientific destiny of our civilisation. So, there are plenty of important similarities, which shouldn't be a surprise, given that Johnson was also the lead designer of *Civilization 4,* but *Old World* feels like a meaningful evolution. And a very welcome one.

Something old, something new

The most notable advance is the importance placed on people. Like *Crusader Kings,* we are not immortal rulers leading our people from the birth of civilisation all the way to the space race and beyond. Instead, our famous starting ruler will eventually die. When Philip II left this mortal coil, the empire passed to his son, Alexander, who'd become known as Alexander the Noble, who then left it to his daughter, 40 years later. We've got heirs and succession laws to worry about, as well as a court full of potential friends and enemies.

As the name suggests, *Old World* sticks to antiquity, slowing down *Civilization's* pace to allow characters to make an impression. It would be tough to do that if every turn moved things forward by decades. Our friends and enemies will be hanging around for a long time, provided something untoward doesn't happen to them.

These courtiers, generals and other important folk grow and react, increasing their attributes to become more charismatic or disciplined, and picking up traits like 'wanton' and 'schemer'. They have affairs and illegitimate kids, and can plot the death of others, including the ruler.

Much of this plays out through engaging event pop-ups, giving us multiple options that depend on our traits and resources. This all contributes to a sense that we are truly shaping our own civilisation, and in ways that go beyond changing some stats. Slavery, for instance, crops up in both *Old World* and *Civilization,* but how it impacts both games is quite different. In *Civ 4,* for instance, it lets you sacrifice part of your population to rush a construction project. In *Old World,* however, it becomes a topic that's up for debate. We might find our people have come to their senses and want to abolish it, encouraging us to toss out the grotesque institution.

Alone, the events and characters would have been enough to grab

Pericles the New is keeping the locals busy. These images are not to scale, we're told.

our attention. *Civilization* and *Crusader Kings* are two of the greats, and combining their philosophies leads to a game that feels tailor-made to our interests. But *Old World* is filled with new ideas and ways of doing things, dramatically changing how we forge a lasting empire.

Every unit has a specific movement range and room for one action per turn, but without Orders, they can't do anything. Orders are a resource that we spend on giving units and characters – like our ambassador or spymaster – commands. Move here. Attack this loser. Steal research from this dork. As we start expanding and fielding more troops, workers and missionaries, we find ourselves having to prioritise where to spend Orders each turn so we don't find our tank is empty right when we're about to make a critical move. It's a strategic complication, but it can take the pressure off. The system teaches us that it's OK to not take action this turn. We don't need to do everything in one go – we've got 200 years to kill.

Orders are tied to Legitimacy, with higher Legitimacy generating more Orders per turn. It's another abstract resource, spawning yet another wrinkle. See, we improve the legitimacy of our reign by being insular, promoting national unity and the people of our chosen civilisation above all else. Often, this means we have to treat foreigners with suspicion and take a strong stance that makes the rest of the world rightly think we're an idiot. It reflects how real leaders sometimes fan the flames of national fervour and encourage small-minded attitudes.

We can, thankfully, still be an extremely effective ruler without pandering to the worst aspects of national identity. Some buildings – which are constructed by workers rather than from the city menu, letting us embark on several construction projects at once – generate small amounts of Legitimacy that, over time, give us a significant boost. That's something *Old World* excels at: giving us more ways to achieve objectives. Resource management is another example. On top of the abstract stuff, there's also more tangible resources, such as stone, iron, wood and gold, which are required for trade and construction. It's a hungry game that demands a healthy stockpile. If we're running low, however, we can simply spend gold to buy more; if we're out of gold, we can sell whatever resource is abundant.

Friends with benefits

When our larder is looking sparse, we can also seek help from our fellow rulers. The focus on people benefits the diplomacy system, as these are personal relationships that we're developing. Friendships with foreign leaders can be completely undone by the way an event plays out, but as always there are plenty of opportunities to repair the damage. More so later on, however, once we've unlocked the ability to employ an ambassador. Or we can use espionage to steal from them. Sometimes the prerequisites feel restrictive, though, like alliances only being possible if we have a diplomatically inclined leader.

Less of a tech tree, more of a slab selection.

If diplomacy fails, we might find ourselves heading to war. Combat is one of the places that feels most evocative of modern *Civ*, which is unfortunate, because we're getting bored of moving all these units around one by one, surrounding cities and slowly battering enemies. That's not to say it hasn't been improved, mind. For one, there's the undo move (or turn) ability, which is a blessing.

The AI is pretty good, too. A bit too risk-averse on the default difficulty, maybe, but smarter and more reactive. Enemies retreat to heal up, take advantage of our weaknesses or injured units, and are less likely to be baited into traps. We're not talking tactical geniuses, but logical, rational enemies are still a boon. Unfortunately, it's still not great at capturing cities. It's a bit slow and sometimes underprepared, but it gets there eventually.

It's a shame it peters out towards the end. The victory conditions suck and mostly go unexplained. The primary way to win is to earn 52 points before 200 years have passed, which you do by nurturing cities and building wonders. If nobody gets to 52, the highest wins. You can wipe out your opponents or claim an Ambition victory. It's a bit perfunctory and more gamey than we expected.

Despite its understated victories, *Old World* is a brilliant 4X, and one that we'd actually recommend over *Civilization* at the moment. *Old World* is full of evocative orchestral and choral compositions. There's a lot to love here, and if you've got an itch to conquer the ancient world, you absolutely need to give this a shot. LXF

Learning to keep it in the family takes on a whole new meaning.

VERDICT

DEVELOPER: Mohawk Games
WEB: https://mohawkgames.com
PRICE: £34.99

GAMEPLAY	**9/10**	**LONGEVITY**	**9/10**
GRAPHICS	**9/10**	**VALUE**	**8/10**

A clever spiritual successor to *Civilization* full of both novelties and big ideas.

» Rating 9/10

Roundup

Akkoma » Misskey » Lemmy » Mastodon » Pixelfed

David Rutland was using the fediverse before it was cool. If it ever becomes popular, he'll go elsewhere.

Fediverse servers

Looking to create your own online community? The fediverse is for you. **David Rutland** tests the best self-hosted fediverse servers available.

HOW WE TESTED...

The fediverse is all about choice. You can pick a server, decide you don't like it, and go elsewhere. If you're running your own server, it's a little more difficult, as you're also responsible for your users, and any changes will affect them, too. It's important to get it right and to choose an instance type that suits your users and your own responsibilities as an admin.

We tested based on usability from these perspectives, and deployed instances on a variety of VPS setups, along with a Raspberry Pi and a more powerful physical server. While we did load up our instances with virtual users to test moderation, threaded conversations and image uploads, we didn't always let them out on to the internet or federate away from our network. For some of the screenshots, we joined large public servers.

For the mobile apps, we used the Android versions from F-Droid, as the author has neither an iPhone, nor a Google Account.

The social media landscape is a mess right now, with platforms changing ownership, changing rules, popping into existence and disappearing.

The newest kid on the block is Facebook's Threads, but who can forget Google's social media ventures, which included Buzz, Friend Connect, Orkut and Google Plus? The social media formally known as Twitter is almost unrecognisable, and only Facebook has survived in a form close to its original vision. But Facebook is old and uncool.

The fediverse is a social network made of tens of thousands of independent social networks – each with its own rules, admins, moderators and peculiar ways of working. Each fedi server can have hundreds of users, or as few as one, and members on each one can talk and interact with users on the others – assuming the admins allow federation.

Some are rock-solid pillars of the ecosystem, while others are ephemeral and will disappear overnight.

Unlike with traditional social media, it doesn't matter if an admin decides to shut up shop, as you can export your profile, along with followers and posts, and join another instance – picking up where you left off.

There are dozens of types of fediverse servers, all designed with a particular aim in mind. Before you spin up your own instance, here's what you need to know.

CREDIT: iStock/Getty Images Plus/Rawpixel

Ease of deployment

How quickly can you set up a server?

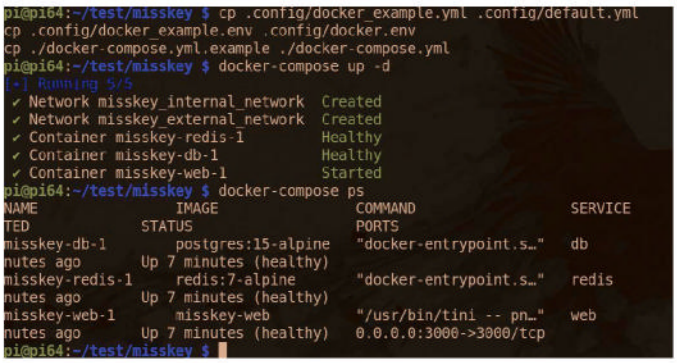
Installing software on Linux isn't always the easiest, and if it was, magazines such as this one would have a hard time justifying their existence. If your aim is to deploy and run your own social media server, you don't want to waste time going through a complicated installation process, trawling through GitHub issues, or following a step-by-step guide to compiling and configuring multiple components from source.

Ideally you should be able to get your fediverse server up and running in under an hour – or an afternoon at most. Complicated setups are a definite disadvantage if you want to start getting social in a hurry, while a one-liner install script would be ideal. In testing our five fediverse servers for ease of setup, we count the time and complexity of getting the server and web interface up and running on a local network. You need to take additional steps to get it working behind a reverse proxy, secured and open to the web. The additional steps are fairly standard, and involve creating a CONF file for Apache, and running *Certbot*.

There are official and unofficial ways of deploying each instance, and if there was an easy deployment solution available, we were willing to dive down a rabbit hole of third-party tutorials.

Mastodon and Akkoma came out very well, with both offering *Docker* installation out of the box. Assuming you have *Docker* and *Docker Compose* already installed…

Installing Akkoma, Lemmy, and Misskey on a VPS was simplicity itself, and after cloning the GitHub repository, we needed to do very little else to generate and bring up the *Docker* containers using *Docker Compose*. Akkoma was especially easy thanks to the instance generation script, which asks a handful of commonsense questions about the server. For all three, we could access our shiny new fedi server within 20 minutes of starting.



Docker Compose makes deploying self-hosted software simple. With a few commands, you can have an Akkoma or Misskey server up and running in minutes.

Mastodon was slightly more difficult – despite using *Docker* and *Docker Compose*, it involved creating multiple directories, generating secret keys and configuring environment files. Still, we were proud to see our instance front page in around an hour.

Pixelfed was the most time-consuming, as there isn't an official *Docker* image. You need to set up and migrate databases, configure PHP, and more. That said, it's still very doable.

VERDICT

AKKOMA	**9/10**	**MASTODON**	**7/10**
MISSKEY	**8/10**	**PIXELFED**	**5/10**
LEMMY	**8/10**		

Setting up Akkoma is an absolute breeze, and none of the other Docker installs were difficult either.

Hardware requirements

Will it run on your old box?

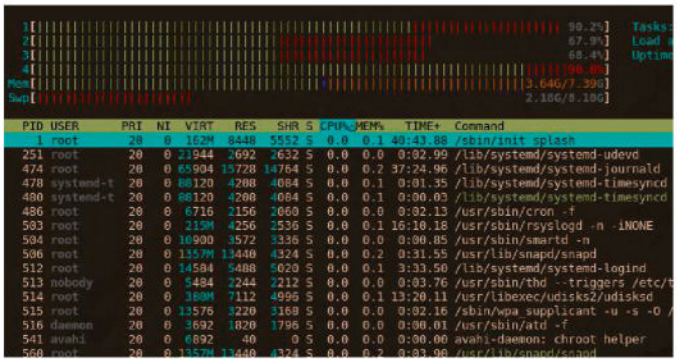
Linux is justifiably famous for running on anything from a pregnancy test upwards, but more complex software requires more powerful hardware, and the architecture of your machine matters, too. And just because you can get your fediverse software to run on your ancient Pentium Core Duo with RAM measured in megabytes, it doesn't mean it will be usable.

For many readers, a Raspberry Pi is ideal for hosting your fediverse server, thanks to its low price and running costs. But this does limit you in terms of the fediverse servers you can easily run.

Mastodon, Misskey, and Akkoma support ARM64 architecture out of the box, complete with *Docker* images. Lemmy does support ARM64, but for *Docker* images, you'll have to check out the unofficial masquernya/lemmy-arm64 repository. We were unable to get Pixelfed running on our test Pi.

Mastodon requires at least 2GB RAM to run, although with this spec, it's a little clunky, and with more than a handful of users, you'll need to scale up your hardware pretty quickly.

We found Akkoma to be the least resource-hungry setup, and were able to get it running on our $10-per-year single-core VPS with 750MB RAM. We probably wouldn't trust it with multiple users, though. Lemmy, Misskey and Pixelfed fall somewhere in the middle. While they failed on our super-budget setup, they ran reasonably OK on a 1GB VPS, making heavy use of swap.



Most fediverse servers work brilliantly on a Raspberry Pi – even when it's also running dozens of other self-hosted services.

VERDICT

AKKOMA	**9/10**	**MASTODON**	**5/10**
MISSKEY	**7/10**	**PIXELFED**	**7/10**
LEMMY	**7/10**		

While all server types can run with limited resources, Akkoma works on machines with minimal specs and at minimal cost.

Customising and special features

What is it that makes these fediverse server types stand out above the rest?

The fediverse is all about doing things your own way while maintaining compatibility with other fediverse servers. Unlike homogeneous services such as Twitter/X, each fediverse server type has a different purpose. This can be to facilitate discussions, publish long-form articles, showcase images, or share your communal love of music. Alternatively, it can be to cram thousands of users into a virtual room and let them scream at each other.

Before you set up your instance, you need to decide what type of community you want to build and how your users will interact with each other. If you're building around a common interest, what special features does each instance type have that differentiates it from the rest?

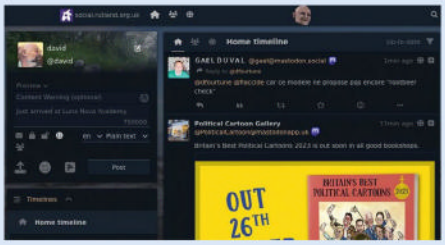
An instance geared towards long-form threaded discussions doesn't cut it if most of your users are there to show off pictures, while a rigid appearance will frustrate those who crave expression.

Akkoma 9/10

Akkoma is this writer's preferred solution – offering a balance of straightforward microblogging and customisation. Use the default Pleroma front-end and the view is easy to understand and doesn't feel cluttered, or switch to the Mastodon FE and it looks just like Mastodon. Changing Akkoma's UI in meaningful ways further than slotting in alt front-ends is a little more tricky once you get past the background and icon stage, though.

The software does borrow some design cues from Misskey, including the ability to see what server type posts on your timeline were made with.

While most fedi servers have a character count of a few hundred, Akkoma's default is 5,000, and you can easily increase this to whatever you want through the admin interface. Want to publish an essay or a book? You can. Akkoma works well for small to middling size instances, and provides a homely feel.

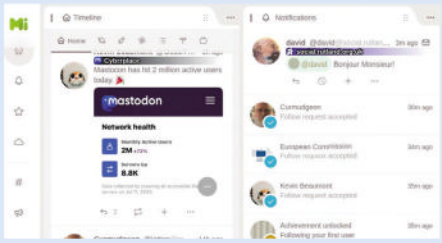


Misskey 9/10

Misskey gives you the ultimate outlet for creative expression on a social network and, honestly, it's a bit shocking just how much you can change, customise and otherwise sculpt to your satisfaction from within the interface.

Using Misskey is like using the internet in the mid-2000s, in the early days of Web 2.0, when there was a bunch of new technologies available and no one was sure what to do with them yet. Wild conceptual designs proliferated on private blogs, major websites and band homepages. If you long for those bygone days, Misskey is exactly the type of fediverse server you want.

Misskey also encourages members to use it as cloud storage and for hosting large files, like Dropbox or Nextcloud. Bear in mind that you'll be deploying Misskey on your own hardware, and SSDs fill up fast when you have multiple users. You may want to turn this feature off.



Customisation options

Make your instance truly unique.

Hark back to the early days of social media, and you'll recall just how wild it was. While we're now used to the standard uniformity of Twitter/X and Facebook, it's easy to forget how pioneers such as MySpace sold themselves on how crazy you could make your page.

When you're designing your own little corner of the internet, you want it to look nice, and you want your users to feel that they can make it their home. If they can't, they may look elsewhere.

Misskey is the standout winner, as it enables you to customise virtually every aspect of your instance's appearance to suit the theme of your server. Although it comes with Japanese catgirls as standard, there's nothing stopping you from changing the theme, icons, layout, avatars and every other detail.

If it isn't customisable enough for you, you can download plugins that extend your options. Users can change their UI, too – adding clocks, widgets and more to the timeline.

Akkoma customisations aren't nearly as extensive, and it uses a forked Pleroma interface as standard - making it easy to set your wallpaper, banners, icons and custom avatars. Even better, it supports a variety of front-ends, so if you prefer the look of Soapbox or Mastodon, you can use those instead.

By necessity, and as an image-themed server, Pixelfed needs to conform to an image-orientated layout, while Lemmy, as a Reddit clone, allows more-or-less the same level of customisation you'd have if you started a subreddit.

Mastodon instances tend to be very uniform, and wherever you are in the fediverse, if you're on a Mastodon instance, you'll recognise it instantly.

VERDICT

AKKOMA	8/10	MASTODON	4/10
MISSKEY	10/10	PIXELFED	6/10
LEMMY	7/10		

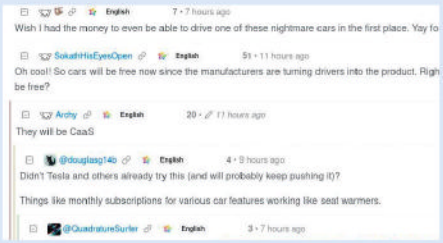
Misskey is the outright winner in terms of user and instance customisation. Add themes, widgets, new layouts and graphs to pretend it's still 2004!

Lemmy 9/10

Lemmy is the most distinctive server type of our fediverse *Roundup*. It's not trying to imitate X/Twitter, Facebook, Instagram or any of the other platforms, where you try to shout louder and get more repeats than any other user. Instead, Lemmy is modelled on Reddit.

Essentially, Lemmy is a link aggregator, which means that users can post a link to a cool page somewhere on the internet, and other users can vote on whether it's worth your time to visit. It doesn't have to be a link, and you can simply ask a question or make a statement if you prefer. Comments and discussions ensue, and both comments and posts can be sorted by Hot, Top, New and Old.

Naturally, because Lemmy is part of a federated system, you can join discussions and vote on remote instances, but its peculiarities mean that it isn't completely compatible with the rest of the fediverse.



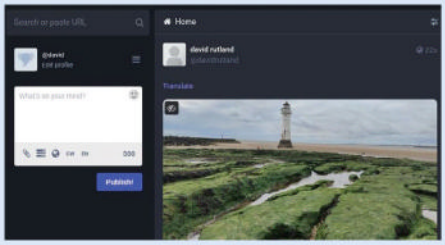
Mastodon 9/10

Mastodon is the biggest name in the fediverse, to the extent that most normies simply refer to the greater network as "Mastodon". This can be frustrating for the rest of us, but its ubiquity is undeniable. There are more than 9,000 Mastodon servers, with a combined total of more than eight million users.

Mastodon is the default for the fediverse, and if you want your prospective users to easily find your server and instantly feel at home if they're moving from a different one, Mastodon's the one you should probably choose.

Aside from its familiarity and ubiquity, Mastodon scales very well, and there are multiple Mastodon servers with a user count in the hundreds of thousands – just add more resources when you need them.

Key features include toot editing, straightforward moderation, and a friendly interface for beginners, showing local, federated and custom timelines.



Pixelfed 9/10

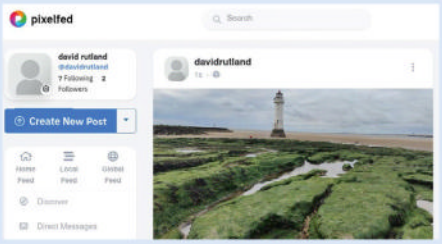
Time was when you saw a cute animal or your dinner arrived with some expert level of preparation and ultra-rare ingredients, you'd snap a shot for the 'gram.

But as a privacy-concerned individual, you may be uncomfortable sharing your data with Instagram (by Meta). You just want to share images of cool things you've seen.

Pixelfed gives you an Instagram-like experience without the sense of terror that comes from being tracked by a predatory advertising company.

Take pictures from within the mobile app, upload them from your camera or desktop. Vote! Discuss! Add useful and accessible alt text!

You can also create themed collections or even Stories – groups of images that last 24 hours before vanishing into the ether, never to be seen again. Although Pixelfed lacks Instagram's filter features, it's otherwise a slot-in replacement.



Fediverse to go

Social media lives in the palm of your hand.

To run a successful social media empire, you need a service that looks good on the go, and allows users to access their feed in comfort and doom scroll on the train, in class or while on a first date in a fancy restaurant.

Fortunately, there are official and unofficial mobile solutions of variable quality for all five of our candidate instance types. The ActivityPub protocol is standard for all fediverse instances, so in theory, every mobile client should work with every server type.

Mastodon, as the largest and best known name, has its own branded app, featuring dark mode, polls and a collection of cute elephant mascots. It's fine, we guess, and great for scrolling chronologically through posts by people you follow on your home server and elsewhere. For new users in search of servers, you can search through available instances and filter by type – or type in your own server address. If you want the fediverse to feel friendly, non-threatening and familiar, this is the one to go for.

Neither Akkoma nor Misskey have their own dedicated mobile apps, but they're 100% compatible with either the Mastodon mobile app or one of the generic fediverse clients you can download from F-Droid or a proprietary marketplace. Of these, our favourite is *Tusky*, thanks to its ability to follow hashtags, schedule posts, and make announcements, and its superior timeline controls.

Pixelfed's app – as you might imagine – prioritises streams of images and hashtags à la Instagram, and while more limited for discussions, is extremely good.

If you've ever used Reddit, you'll be at home with Lemmy's semi-official *Jerboa* app.

VERDICT

AKKOMA	**7/10**	**MASTODON**	**6/10**
MISSKEY	**7/10**	**PIXELFED**	**10/10**
LEMMY	**8/10**		

While all the mobile apps are good, Pixelfed is best for losing yourself in the mobile feed. Mastodon's app is friendly, but has fewer features than we'd like.

Moderation and safety

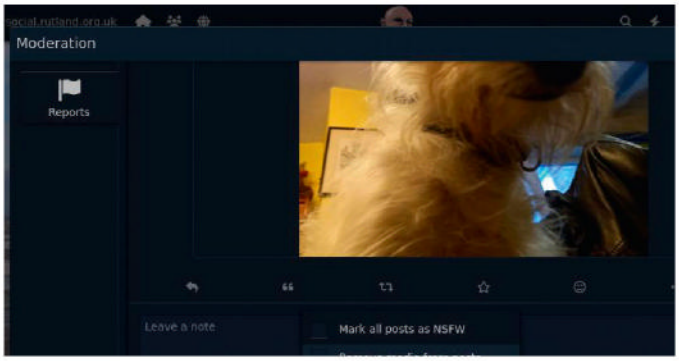
No one wants a server full of filth.

It's a long-standing criticism of large social networks that moderation policies make the experience terrible for everyone. Your own posts get banned for some trivial reason, while libels are spread about you by others. Before you know it, someone invokes Godwin's law and it's all over.

Moderating social media is difficult, and as the admin of your own network, the task falls to you and any volunteer mods you can rope in to scour their eyeballs with bleach on a regular basis.

Moderation is fairly straightforward on all five instance types. It has to be. As the instance owner, you're legally on the hook for illegal content and copyright violations if you don't act promptly. Ideally, users should be able to easily flag posts that violate your terms of service or content policy. We found this process quite intuitive and easy on every instance: click the three dots, click Report, write why you're reporting the post, then submit. Admins receive a notification, inspect the details, and choose to deactivate, delete or place restrictions on the offending account.

The work involved with moderation scales along with your instance size, so it's appropriate that Mastodon has the most granular tools. While Akkoma enables you to mark all of a user's posts as NSFW, force posts to be either unlisted or restricted to followers only, or even automatically strip all media, Mastodon also provides tools to freeze the account, limit it (shadowbanning the user) or suspend it entirely, as a precursor to deletion.



Moderation can be as simple as ticking a box. But then the world wouldn't get to see this delightful photo of a Jack Russell.

Other servers may have admins with different ideologies or who are ineffective. As a last measure, you can defederate from an instance, cutting all ties between your users and theirs.

VERDICT

AKKOMA	8/10	MASTODON	9/10
MISSKEY	8/10	PIXELFED	8/10
LEMMY	8/10		

All the instances had effective and easy reporting and moderation tools. Mastodon stood out due to its extensive and nuanced moderation actions.

Holding a conversation

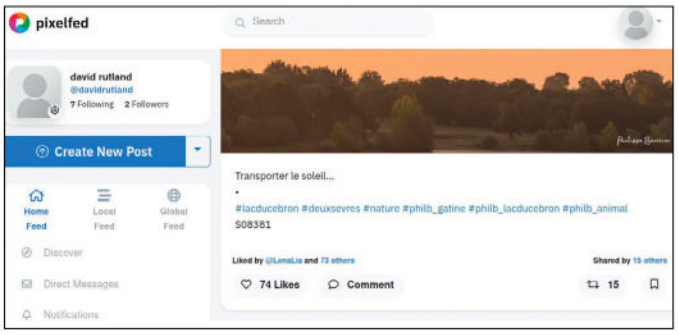
Talking is the best way to be social.

Social media is supposed to be social. It's in the name, and the phenomenon has been compared to a town square or a village pub, where you can grab a pint, sit down at a table, and pitch in with your opinion on the latest football performance (it was a disgrace). Alternatively, you and your mates should be able to sneak off to a private snug and discuss marital woes away from the cacophony of other voices.

The layout of the venue should facilitate both of these conversation modes, while also talking to people in the pub next door through a handy hole in the wall.

Discussion is top priority with Lemmy. It's built around the idea of positing an idea or article, and then talking about it. Conversations are threaded, and each comment can spin off dozens of other comments. While this is true of the other server types, Lemmy's voting mechanism, as well as the ability to sort comments based on votes, age and more, makes it a lot easier to join in the conversation at a relevant point and address the part of the issue you want to talk about.

At the other end of the spectrum is Pixelfed. As with all of our test servers, you can do the equivalent of liking, sharing and commenting, but in practice, this doesn't tend to happen. On Pixelfed servers in the wild, it's typical to see an image liked dozens of times, shared hundreds, and with zero comments. We couldn't say why – there is, after all, a clickable speech bubble and comment field. It just doesn't seem to have been given much priority in the design phase, and it actually feels weird to comment or discuss.



An image with 74 likes and 15 shares should feel like a triumph, but without other engagement or comments, it's a bit hollow.

The Mastodon web interface is an information overload, and you're likely to have three or four streams of posts open at any time. You'll see the comments as individual posts, then expand an interesting tidbit to view the entire conversation. Navigating branches to find the appropriate point at which to engage isn't always intuitive, but it's fun, and you rarely feel you're butting in inappropriately. Akkoma and Misskey are just a fun experience.

VERDICT

AKKOMA	8/10	MASTODON	9/10
MISSKEY	8/10	PIXELFED	4/10
LEMMY	9/10		

For facilitating structured conversations you can jump right into, Lemmy makes it easy, and you'll rarely find people picking a fight by the pool table.

The Verdict

Fediverse servers

This is going to sound like the kind of cop-out you'd hear at a primary school sports day, but there's no clear-cut winner here.

We picked these particular five fediverse servers because they're so very different. They're designed for different types of user, different use cases, and for admins with varying levels of expertise.

While Akkoma is this writer's own personal favourite, we can't recommend it to everyone, and it's best used as a single-user instance or with up to a few dozen local users. The interface, although switchable, is best suited to informal conversations and captioned images.

Likewise, Lemmy is better suited to debates and discussions, and does a fair job of imitating Reddit. However, outside this use case, it's not really the best in any other area.

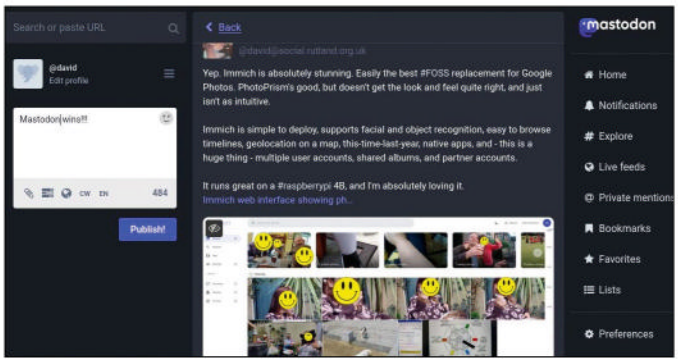
This specialisation is a problem for Pixelfed, too – but only if you're measuring against metrics that are outside of its core function. It's an Instagram clone, and a very good one. To our mind, the ability to federate with different server types is a secondary concern at most, and comments and discussion are largely irrelevant. It's not really fair to judge it on any grounds apart from snapping, sharing and viewing pictures.

Misskey can be whatever you want it to be, and if you're the kind of person who enjoys spending days ricing your desktop to achieve the perfect aesthetic, you're probably going to love it, but most people would possibly prefer fewer options, and just boot the server and get posting.

Mastodon achieves just that. There's a standard interface that is recognisable to approximately seven million users who are already on the fediverse, along with a suite of tools that make sense, and an interface that allows some degree of customisation, but not so much it'll swallow you up.

Taken objectively as a whole, Mastodon comes out ahead, thanks to its sensible controls and extensive moderation options, and for the fact that if you tell your friends to join Mastodon, there's a chance they might actually know what you're talking about.

The other four entries come joint second. In one way or another, they're all great.



1st Mastodon 9/10

Web: **https://joinmastodon.org**
Licence: **AGPLv3+**
The original classic. Easy to use and with good moderation tools. A little dull.

2nd Lemmy 8/10

Web: **https://join-lemmy.org**
Licence: **AGPLv3.0**
It's just like Reddit, but federated across thousands of servers.

3rd Pixelfed 8/10

Web: **https://pixelfed.org**
Licence: **AGPLv3.0**
If you love Instagram, but want to host it yourself, choose Pixelfed.

4th Misskey 8/10

Web: **https://misskey-hub.net/en/**
Licence: **AGPLv3**
Come for the catgirls, stay because you're addicted to tweaking every detail.

5th Akkoma 8/10

Web: **https://akkoma.social**
Licence: **AGPLv3**
A small, cosy instance in which you can relax and write million-word essays.

» ALSO CONSIDER

An even smaller federated social media service you could consider is Diaspora*. It has been around since 2010, making it a sort-of uncle to the fediverse.

It operates in a similar way, with thousands of federated servers supporting text, images and videos, but it doesn't use the ActivityPub protocol that really kick-started the fediverse in 2018.

As a result, the Diaspora* network is home to only around 800,000 people. It's not dead, though, and if you visit any Diaspora* 'pod', you'll see lively discussion between members on the home pod and federated elsewhere.

Development isn't as fast on the fediverse, and there isn't the same degree of server specialisation – either in niche common interests or in how the servers function. So, if the profusion of fediverse server types is giving you a headache, you might consider Diaspora* as your next federated online home.

STREAMING MADE EASY!

Bring together all your film, TV, music, audiobook and photo streaming needs under one roof. **Nick Peers** shows you how.

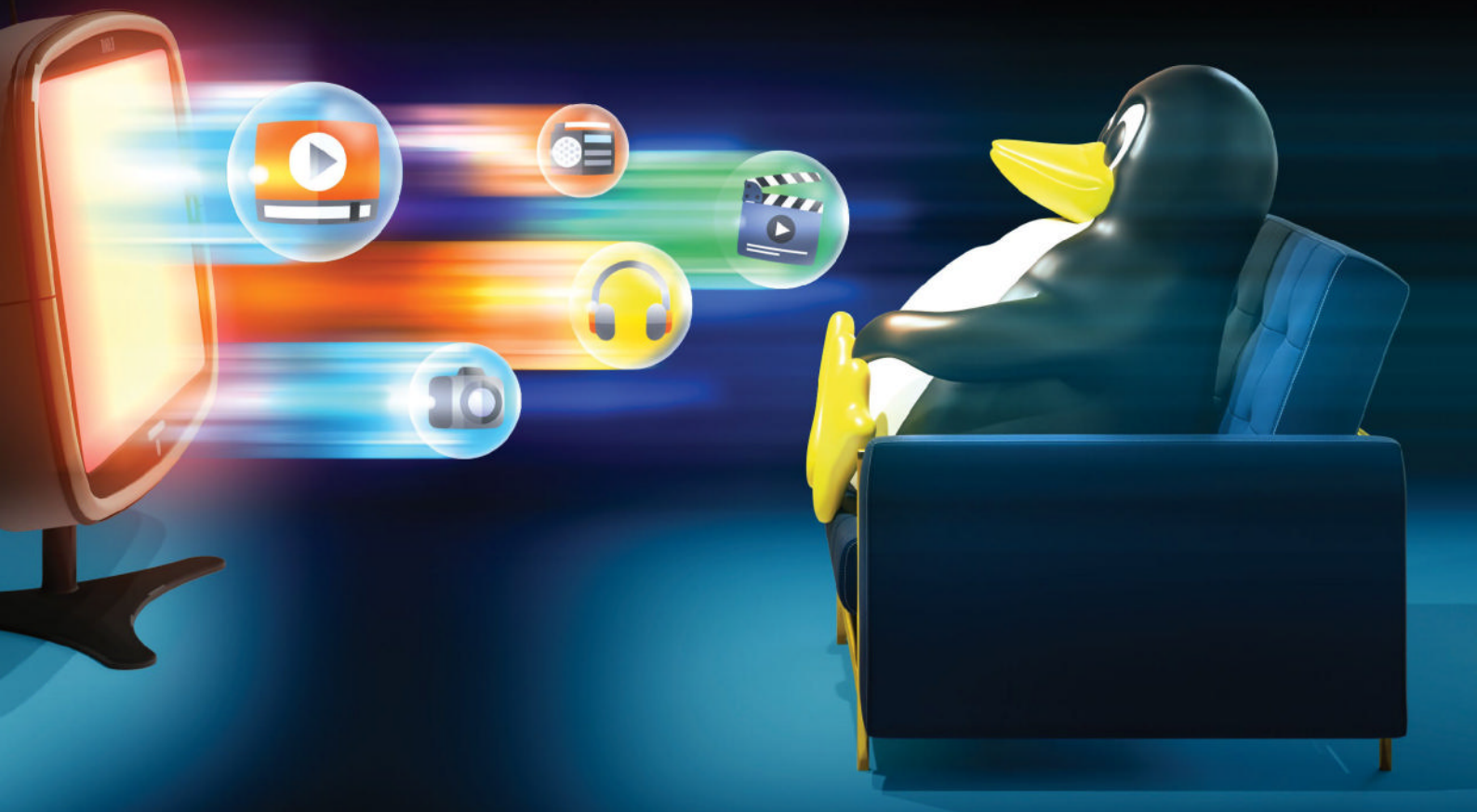
It wasn't that long ago that selling the idea of a self-hosted and self-owned media server seemed like harking back to the days of the dinosaurs. "But everything's on Netflix, iCloud and Spotify!" came the response. But as subscription costs keep rising and services suddenly start dropping content from their libraries, who's laughing now?

There's no better solution to ensuring media is always available to you than by owning it yourself. But why compromise on the convenience of streaming services when your PC gives you the best of all worlds? Thanks to a triumvirate of beautifully written open source tools, you can set up streaming services covering all your media – from music and movies to audiobooks and even all those photos on your phone – without having to pay a penny in subscription fees.

We'll run you through the installation and setup of *Jellyfin* – a feature-rich media server that more than holds its own against rivals *Plex* and *Emby* (never mind Amazon Prime and the rest), with everything you need to build a video and music library packed with content that involves no intrusive ads.

Then there's *Immich*, a new way to back up your photos and view them a variety of ways through your web browser or mobile – by date, face or place. And last, but not least, the brilliant *Audiobookshelf* does more than provide you with a free alternative to *Audible*; it can even handle ebooks and podcasts as well as audiobooks.

Ready to cast off big business and regain control of all your media? Read on, MacDuff...



Install Jellyfin

Get set up and ensure your media files are organised correctly.

There are multiple ways to install *Jellyfin* on your server. You'll find instructions across a range of distros at **https://jellyfin.org/docs/** – taking Ubuntu as an example, open a terminal window and issue the following commands:

```
$ wget -O- https://repo.jellyfin.org/install-debuntu.sh
$ sudo bash
```

This launches a script to add and configure the *Jellyfin* repository, install *Jellyfin* and configure it to start automatically with your server and run in the background continuously. Verify the server is running by opening your web browser and navigating to **http://localhost:8096** or, if you want access from another computer on your network, **http://192.168.x.y:8096** (**192.168.x.y** is, of course, your server's IP address). When you see the welcome wizard, close your browser, then reboot your server and connect again to verify the service is configured correctly.

Prepare your media libraries

Once *Jellyfin* is verified as up and running, but before you start setting up, make sure your media libraries are ready to go. You'll want to set up individual folders for each library you plan to create, which typically means separate folders for Movies and TV, plus one for Music.

Inside each folder, you also need to ensure your files are both organised and named correctly following these conventions:

Type	Folder structure	Syntax	Example
Music	Music\Artist\Album	00 - trackname.mp3	Music\Blur\The Ballad of Darren\01 - The Ballad.mp3
Movies	Movies\Title (year)	Title (year)	Movies\Oppenheimer (2023)\Oppenheimer (2023).mp4
TV shows	TV\Show Title\Season	TVshow – s01e01	TV\Star Trek - Picard\Season 3\Star Trek Picard – s03e01.mp4

If your files aren't currently named correctly, *RenameMyTVSeries* makes short work of renaming files using an online search tool to match episode titles to files. Go to **www.tweaking4all.com/home-theatre/rename-my-tv-series-v2** and download the latest stable Linux version, then extract its contents to a suitable folder. Before double-clicking the *RenameMyTVSeries* shortcut, you may need to install additional dependencies:

```
$ sudo apt-get install libssl-dev sqlite3 libsqlite3-dev
```

When it comes to renaming music files, try *MusicBrainz Picard* (**https://picard.musicbrainz.org**), which can be installed via Snap or Flatpak as well as through its own repository.

SERVER SYSTEM REQUIREMENTS

You're installing three tools to meet your needs, but at the heart of your server is *Jellyfin* (**http://jellyfin.org**). It enables you to bring together all your owned media in one place that can be accessed from just about any device, including Roku, WebOS and Android TV.

In an ideal world, you'd install *Jellyfin* on a server that can be left on 24/7 without guzzling huge amounts of electricity. If you plan to stream to multiple devices at once, it's a good idea to choose a server with a modern GPU that can handle hardware transcoding without compromising on video quality. Consider the following entry level: Intel Graphics HD 600, Nvidia Turing or AMD Vega GPUs.

Sourcing a suitable GPU enables you to get by with a less powerful CPU – Intel does a nifty line in low-powered CPUs, including the Celeron J4125B and Pentium Silver J5040 – ASRock's J5040-ITX motherboard/CPU combo can be had for under £140. Just add 8-16GB RAM, storage and a suitable case. Speaking of storage, you need plenty if you're streaming lots of HD content – 4TB or larger, but leave room to add more, either via USB or by housing your server in a case that can handle multiple internal drives, such as the Jonsbo N2 (£140, **www.overclockers.co.uk**). We also recommend choosing dedicated NAS drives like the WD Red or Seagate Ironwolf series.

Once your media files are in place, it's time to get set up. Open your browser and return to the *Jellyfin* wizard, then work your way through it. It's largely self-explanatory: set yourself up as the main admin user – but don't leave the password blank, particularly if you plan to open up *Jellyfin* for remote access. Next, click Add Media Library to add your first library: select its type (choose Shows for TV), give it a suitable name and then add the library's parent folder before running through the various library options – note, you can configure these later from *Jellyfin's* dashboard if you're impatient to get going. Click OK when done, then repeat for any more libraries you want to add (again, these can be added later) before clicking Next.

Confirm your metadata language and country settings, and click Next. The final step confirms whether you want to open remote access to your server (for streaming content over the internet) and whether you'd like to attempt UPNP port mapping. Once configured, click Next followed by Finish, and *Jellyfin* starts scanning your media folders for content in the background. This can take some time depending on your server hardware, so now's a good time to leave *Jellyfin* to its own devices for a while.

QUICK TIP

If you want to create a reverse proxy, find our full guide here to download: https://linuxformat.com/files/pdfs/LXF306.tut2_proxy.pdf

CREDIT: Magictorch

Stream films, TV and music

Jellyfin can cover almost all of your media streaming needs...

Once *Jellyfin* is set up, you're taken to your home screen, which offers a variety of ways to view your media library: at the top are shortcuts to individual libraries, followed by the Next Up section, which helps you track your progress through TV shows by helpfully displaying partially played episodes as well as queuing up the next available episode for viewing. Beneath this you'll see the latest additions to your libraries.

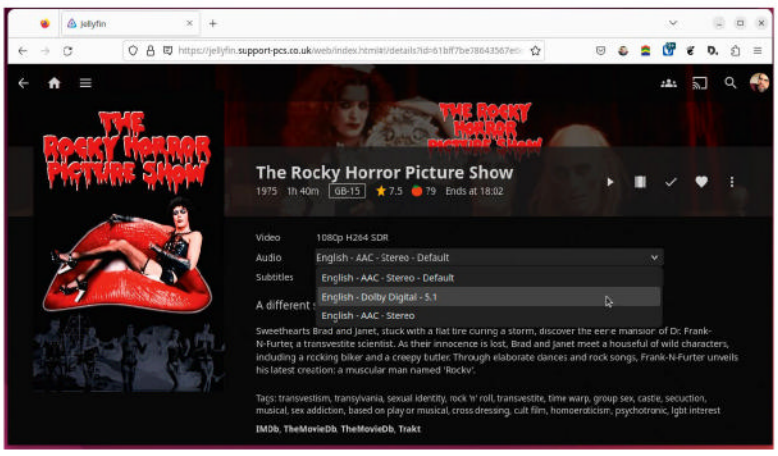
You'll find plenty of on-screen aids to help you navigate even the largest libraries. At the top are a range of library views, from Suggestions to Genres. Look out, too, for Favorites – which lists any media you've marked by clicking the heart icon next to it – and Collections, which enables you to group movies by your choice of criteria, such as franchise or lead actor.

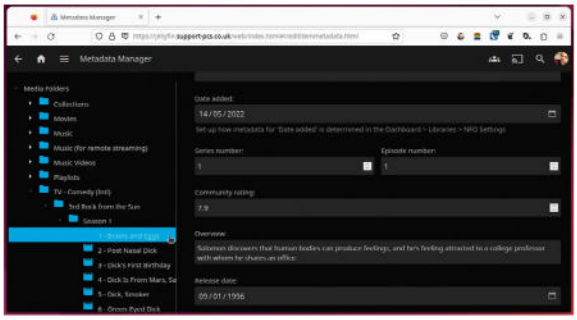
Below are three buttons, enabling you to change from the default poster-based view to show banners, lists or thumbs. Next to this is the A-Z button, enabling you to sort your library by different criteria, while the final button allows you to apply filters to the view, from features and genres to parental ratings and years.

When you select a specific movie, TV episode or other media, you're whisked to its own page, where you'll hopefully see a synopsis as well as a summary of the media's audio and video type. If the media item contains a choice of video, audio or subtitles, look for drop-down menus enabling you to switch from the default choices before playing your movie.

If you've gone the extra mile to add supplements to your media library, such as featurettes or deleted scenes, you'll find these here, too – embedded alongside the movie, or at season level with TV shows. See **https://jellyfin.org/docs/general/server/media/movies** for more pointers on how to name and organise these in your libraries.

Jellyfin provides a wealth of information about the content in your media server – as well as playback options.





Jellyfin does the legwork of importing metadata into your server, but you can still edit individual aspects of it.

Fix metadata

As you browse your library, you're likely to come across the odd series or movie that has been incorrectly identified. To fix this – plus make other metadata-related changes – click the hamburger (≡) button in the top-left corner of the screen to slide out the main *Jellyfin* menu and choose Metadata under Admin.

The Metadata Manager reveals an Explorer-like view of your media by folder. Drill down to a movie title, or parent folder for a TV show, and you'll see its metadata in the right-hand pane. You can manually edit this if you wish, but if you're trying to fix an incorrect match, a quicker method is to click the vertical ellipsis button next to Save and choose Identify.

This brings up a search tool – you can search by title and/or year (which should resolve most issues where an identically titled show or movie from a different year has been selected), but if this doesn't yield a result, go online to **www.imdb.com** or **http://thetvdb.com** to manually locate your show and grab its ID. Enter this into the relevant search field and you're guaranteed to get the correct match when you click Search.

Manage your server

To configure your server, open the main menu and choose Dashboard under Admin. There are four main sections: Server, Devices, Live TV and Advanced, with Dashboard at the top. It provides a handy overview of recent changes, which devices are connected and what content is streaming – and who's streaming it.

Beneath this, the General section enables you to change your server name and preferred language, followed by Users. If you're planning on sharing access

to *Jellyfin* with other users – both locally and remotely – set up individual accounts for each person with their own custom access to your libraries. In return they gain their own personalised user experience, including their own Up Next deck. Simply click + to set up your first user: assign them a name and password, and choose what libraries to give them access to. Click Save.

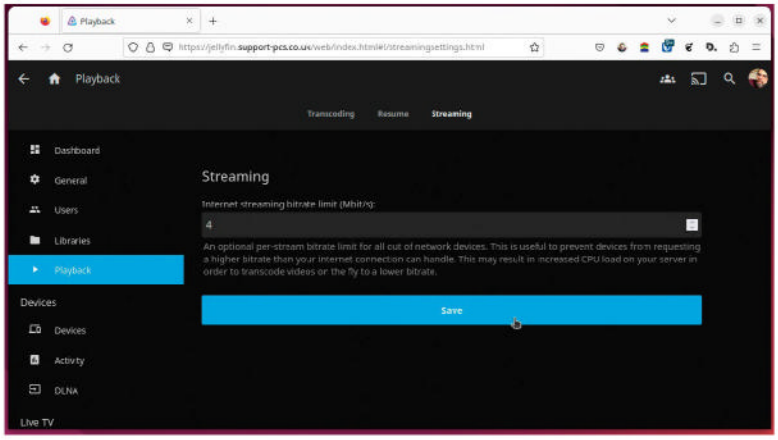
Once a user is created, you gain access to four tabs of settings: Profile, Access, Parental Control and Password. Under Profile, you'll see options for enabling or disabling remote streaming, management of the server and access to specific features, such as Live TV, SyncPlay and media downloading. Access is where you manage their access to both libraries and from specific devices, while Parental Control gives you handy control over children's access to content on the server, with options for setting a minimum parental rating, plus limiting access by day and time. Use the Password tab to assign a new password or add a PIN code for easier access. Click Save when you're done.

Libraries is – of course – where you go to manage existing libraries, plus add new ones (the process is identical to that in the first-run wizard). The final setting under Server is Playback. Available options here include whether to use hardware acceleration for transcoding if supported – remember, there may be a price to pay in terms of image quality, so experiment with toggling it on or off. You'll also find advanced conversion settings for transcoding to x264/x265 – *Handbrake* users should recognise most of these options, and this flexibility is something that sets *Jellyfin* apart from other media servers, such as *Plex*.

If you're planning to access your server remotely (or share it with others), you need to configure *Jellyfin* accordingly. While its Advanced > Networking section enables you to do this manually, we recommend setting things up through a reverse proxy, such as *Nginx Proxy Manager* (which we featured in **LXF306**).

First, configure *Jellyfin* – because the reverse proxy is doing all the hard work, all you need to do is ensure Allow Remote Connections To This Server and Enable IP4 are both ticked under Advanced > Networking.

Now get yourself a suitable domain, subdomain or free dynamic domain, as outlined in the *Nginx Proxy Manager* tutorial, make sure it's pointing to your public IP address (as revealed at **www.whatsmyip.com**) and then click Add Proxy Host in the *Nginx* web interface.



Jellyfin's Dashboard contains access to all the settings and preferences you need to fine-tune it to your needs.

Enter your domain name, leave Scheme set to http and set the Forward Hostname/IP to your server's IP address and the Forward Port to 8096. Switch on Block Common Exploits before selecting the SSL tab. When setting up your free SSL certificate, don't forget to flick the Force SSL switch on to ensure all connections from outside your network to the server are secure.

Manage playback

If you're planning to allow remote access to your server, you also need to be mindful of the speed of your internet connection. Benchmark your connection using a website such as Speedtest (**www.speedtest.net**). You need to compare your connection's upload speeds with the typical bitrates required to stream content to determine how many remote streams your connection can handle simultaneously.

Once done, ration your available bandwidth under Playback > Streaming. Here you can set a per-stream bitrate limit – any media with higher bitrates is transcoded to fit whatever limit you set, so bear this in mind if you're streaming from a low-powered device, and remember that offloading transcoding to the GPU under Playback > Transcoding enables you to handle many more streams simultaneously, albeit with a slight hit to streaming quality.

» CONSUMING YOUR CONTENT

It's perfectly possible to watch and listen to your media through your PC's web browser – simply log into your account and the web interface has everything you need to display photos, play music and watch films and TV shows, but that's rarely your preferred option.

Thankfully, a wide range of apps have been developed to make things easy for you – all you need to supply is the IP address (or subdomain if you configure *Jellyfin* for remote access), along with your username and password. To see what platforms are supported, visit **https://jellyfin.org/downloads**, where you can find official *Jellyfin* apps for Apple and Android phones and tablets in their respective stores, alongside *Jellyfin Media Player* for desktop. *Jellyfin* also shows up in WebOS, AndroidTV and Roku stores, too, while both mobile apps also support Chromecast, should your TV not be directly compatible.

When browsing clients on the *Jellyfin* website, flick the switch to All and you'll reveal loads of free third-party alternatives, too. One app we recommend looking at if you plan to stream music to your phone or tablet is *Finamp,* which provides a more optimised experience for audiophiles.

Note, there's no centralised developer of *Jellyfin* apps, so expect a marked difference between them – particularly regarding development. One example of an app receiving a lot of attention is the Roku app, which after a slow start after its initial appearance in 2020 has undergone rapid development, with a slew of releases in 2022 and 2023.

Back up and view photos and videos

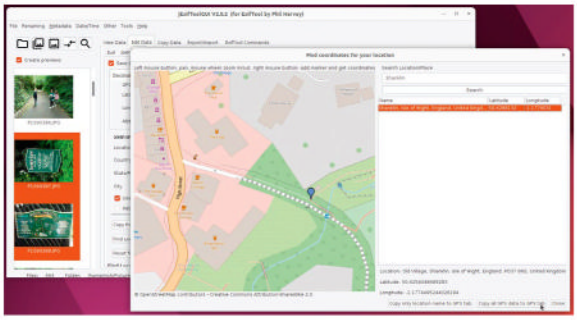
Ditch the expensive subscription and back up (and stream) photos locally.

While it's possible to create libraries in *Jellyfin* to stream your personal photos and home videos to the big screen, the question remains of how easy it is to transfer said content from your phone to your server. Taking its cue from your cloud provider, *Immich* is designed to provide you with your own self-hosted alternative – one that doesn't come with expensive subscription fees, and which not only simplifies the task of transferring your photos and videos to your server, but also gives you the means of organising files by album, people, location and more.

Immich installation

Immich is supplied as a container, so you need *Docker* or similar to deploy it. Browse to **https://documentation.immich.app/docs/install/requirements** for the scripts and instructions you need – *Docker Compose* is the recommended route to follow, but we installed our instance easily using *Portainer's* stacks feature.

The key environment variable you need to consider is UPLOAD_LOCATION, which is where all photos and videos will be uploaded to on your server. If you plan to create libraries in *Jellyfin* for accessing this content, make sure it can access the path you set here.



Edit your photos' metadata before you import them into Immich – jExifToolGUI offers an easy way to do just that.

It can take five to 10 minutes for *Immich* to set itself up and start, so be patient. Once complete, open your web browser and navigate to **http://192.168.x.y:2283** (where **192.168.x.y** is your server's IP address) to get started proper. The first thing is to set up the administrator account – you need your email, a strong password, and first and last name. Click Sign Up and you're prompted to log in with your new credentials.

Get started

You'll find yourself at the main web interface, ready to upload your first photo or video. Click the Upload button or drag your file(s) into the browser window and wait while they're uploaded before refreshing the page to see the files pop up. They're filed by date and clicking one shows it in full-screen with buttons across the top enabling you to view at full size, download back to your PC, view metadata, favourite the photo or delete it. Clicking the vertical ellipsis button on the right reveals more options, such as adding the photo to an album (or shared album), archiving it, setting it as your account profile picture and refreshing metadata and thumbnails.

Click the 'i' button to view its metadata, which is embedded in the file and should include the date and time the photo was taken as well as its filename, size and resolution. You may also see additional information – geocoordinates if the photo's location was recorded when it was taken, or the camera's make, model and photo settings, for example. You'll also see a description field, which can be edited if you wish to add one.

Upload from mobile

Immich offers apps for F-Droid, Android and iOS – just search the respective store for it. Once installed, launch

» EDIT PHOTO AND VIDEO METADATA

Immich works wonders with existing metadata, such as location information and date creation, when offering you ways of navigating your library, but it has no built-in capabilities of its own. This means it pays to use a tool such as *ExifTool* (**https://exiftool.org**) to manually edit each file's metadata before you upload it to *Immich*. If your file is already on the server, simply download it back to your PC, make your changes, and then upload it again (if necessary, delete the original file from the server first).

ExifTool is very powerful and not too hard to master, but if you'd rather not wrestle with a command-line app, try a third-party GUI like *jExifToolGUI* (**https://hvdwolf.github.io/jExifToolGUI**). Download the DEB package to your PC, then open a terminal, navigate to the directory containing the DEB file and issue the following command:

```
$ sudo dpkg -i jexiftoolgui-2.0.2.deb
```

You'll likely be met by an error about missing dependencies, which can be rectified with the following command:

```
$ sudo apt -f install
```

Once installed, drag a file or files into the program's left-hand pane. Use the tabs on the right to view current data and make changes – you'll find additional tabs enabling you to edit all kinds of data, including GPS and geotagging to help *Immich* correctly place the photo(s) on a map. Visit **https://hvdwolf.github.io/jExifToolGUI/manual** for a complete guide to using the program.

the app and you're prompted to enter the Server Endpoint URL – if you're happy to only use *Immich* while at home, you can simply enter **http://192.168.x.y:2283/api**, or if you'd like remote access, you need to jump through several additional hoops. The following assumes you're adding *Nginx Proxy Manager* to work on top of *Immich's* internal reverse proxy, and that you are managing containers using *Portainer.*

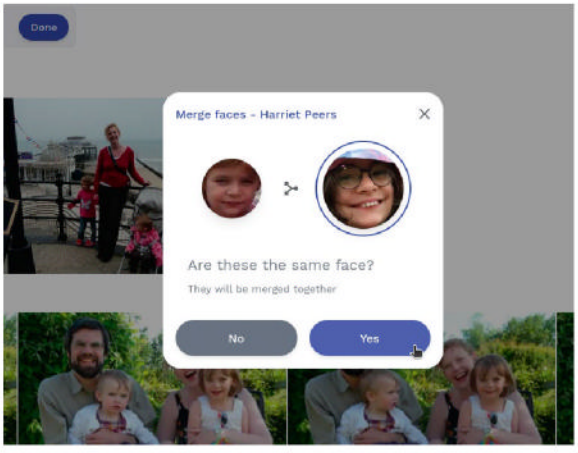
First, log into *Portainer* and click on Immich_proxy, then scroll down to the Connected Networks section and join it to the same bridge network *Nginx* is on, so it's connected to both that and its own internal immich_default network.

Open *Nginx Proxy Manager* and add a new proxy host pointing to your chosen domain or subdomain. Leave Scheme set to http, set Forward Hostname/IP to immich_proxy and Forward Port to 8080. After setting up your SSL certificate for secure remote access, click OK and verify the connection works in your web browser simply by browsing to your chosen subdomain or domain (such as **https://immich.domain.com**). Once verified, enter this URL into the Server Endpoint URL field in the mobile app, then log in as normal.

Give *Immich* the requisite permissions to all your photos and videos when prompted, and you'll find yourself at a similar-looking screen to the main web interface. Start by tapping the cloud backup button in the top right-hand corner, where you can select which albums to back up, then configure automatic foreground and background backing up, so you don't have to manually keep an eye on *Immich* while it transfers your media to the server (look for the handy Only While Charging option to restrict *Immich's* background uploads to when your mobile is connected to the mains). Once configured, click Start Backup and watch while *Immich* begins the process.

Manage your photos

As your collection grows (in real time, if you're backing up from your mobile), you'll see that photos are filed in date order. At some point you'll want to bring some order to this chaos, and that can be done in several ways. The simplest method is to file photos into albums using *Immich's* own tools. Click Albums on the left, then click Create Album to name your first album (and provide an optional description). Beneath this you'll see a Select Photos option. Note that photos can reside in more than one album.



Immich offers built-in facial recognition, and you can merge entries to consolidate photos of a person young and old(er).

You can also tag special photos as favourites, or archive photos that you don't want to appear in the main feed (simply access them through the Archives shortcut under Library).

But *Immich* enables you to go further – click Explore and you'll discover it has built-in facial recognition and tagging options, too, as well as being able to organise photos by place. Let's start with facial tagging. Click View All next to People to see all detected faces, then click one to add a name to the face, plus verify all photos assigned to that face are correct (if any aren't, simply roll your mouse over the photo and remove the tick). If a person appears in this list twice, simply enter the same name again for the second instance, then merge the two together when prompted.

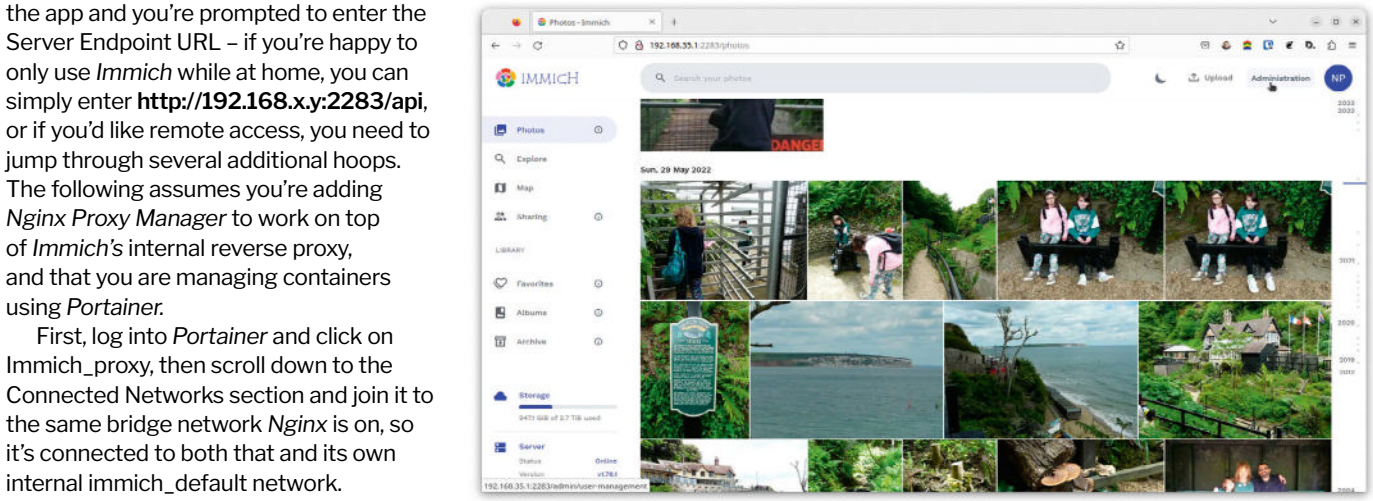
When browsing by place, click on a photo and you're shown its properties screen complete with a handy Place tag under Details, along with a map pinpointing where the photo was taken, which you can zoom right into to pinpoint the location exactly.

Sadly, as things stand, *Immich* has no capabilities for editing metadata such as date and location of photos – check out the box *(opposite)* for a way to fix this manually. Once you're set up, click Administration to further configure the server. You can create other users for family and friends, run jobs such as tagging objects and encoding thumbnails, plus access server stats.

Moving on

Immich is a fantastic self-hosting video and photos backup tool, but what about viewing your media? Both the web interface and mobile app enable you to browse, view and run slideshows, but if you want to go further and view your consolidated home media on the big screen, you need to revert to *Jellyfin.*

Once your photos are uploaded and organised, switch to *Jellyfin* and navigate to Server > Libraries under Dashboard. Click Add Media Library and choose Photos to set up a library pointing to the relevant user folder inside the folder you specified in the UPLOAD_LOCATION variable when setting up *Immich.*



Immich is designed to look – and work – like your phone's existing cloud storage solution, without the subscription fees.

Stream audiobooks from your server

Why pay for Audible when you can host your own collection?

One often overlooked component of a media server is audiobooks. In the past, we've recommended *Booksonic*, but in 2021 a new arrival quickly swept all before it to provide a serious alternative to Audible. *Audiobookshelf* (**www.audiobookshelf.org**) is your open source one-stop shop for self-hosted audiobooks and podcasts, and recent releases have even started to add support for ebooks as well.

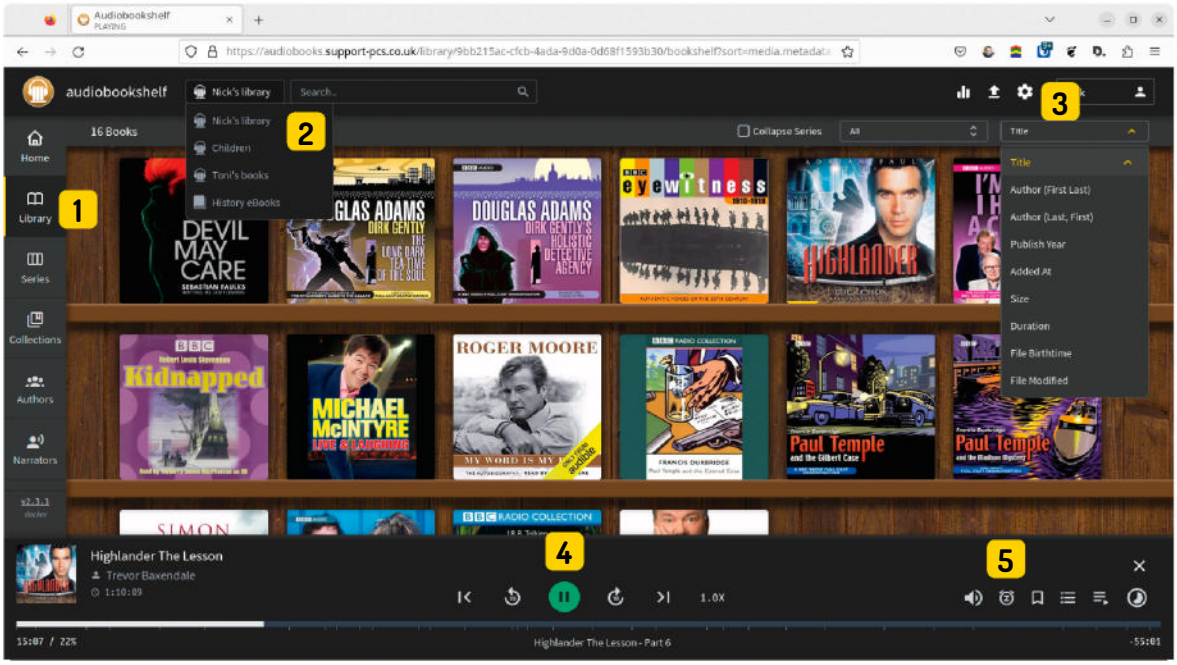
You can install it through *Docker* or natively – simply follow the instructions at **www.audiobookshelf.org/docs/** for your chosen path. The native installation works out of the box, but if you wish to change any defaults – specifically which port it communicates on (13378 by default) – you need to edit the configuration file at **/etc/default/audiobookshelf**.

Should you be planning remote access through *Nginx Proxy Manager*, you'll need to set the scheme (http), forward path (your server's IP address) and port (13378 or whichever port you specify during setup), plus make sure you enable Websockets support.

Initial configuration

You should start by bringing all your audiobooks together in a single location – the box *(opposite page)* reveals where to source audiobooks and how to organise them on your server. If you plan to share your server with others (go to Settings > Users to set them up after installation), they may want to keep their own libraries separate, in which case add another layer of organisation to your folder structure, so it reads User > Author > Series > Title > files.

Navigate your audiobook collection



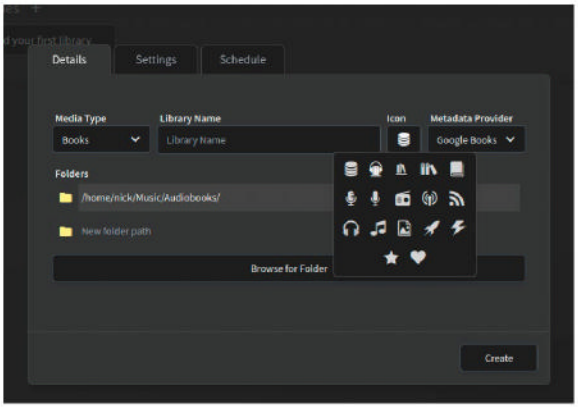
1 View your collection
Use this menu to switch between different views – Home is an overview, or select Library for a complete list by title.

2 Switch library
You can add multiple libraries – each one is self-contained and isolated from the others. Click here to switch.

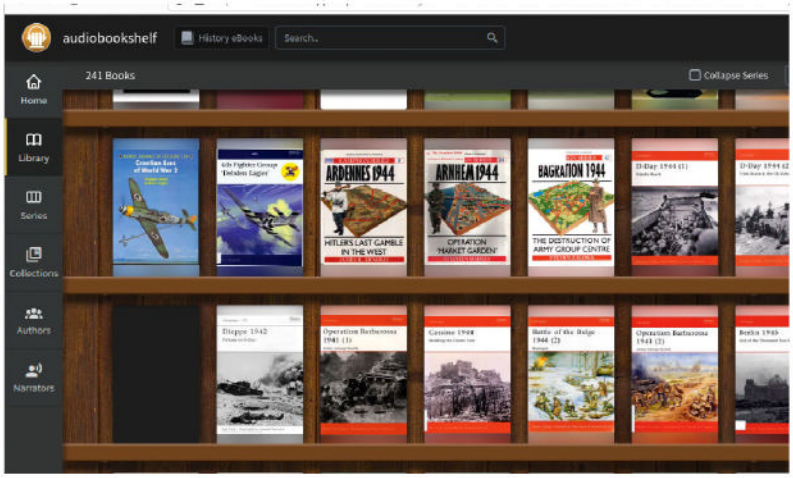
3 Upload new content
You can add new content by copying it into the relevant library folder or by clicking this button to upload individual files.

4 Audiobook player
Stream an audiobook from your browser by clicking it, then use these playback controls – including an option to change speed.

5 Sleep timer
Need a soothing book to send you to sleep? Click the alarm clock to set a timer, after which playback pauses.



Your first job after installing and logging into Audiobookshelf is to create your first audiobook library.



Audiobookshelf can also handle podcasts and ebooks – the latter is still in its infancy, but already well-formed.

Once in place, open *Audiobookshelf* in your browser by navigating to **http://192.168.x.y:13378** or **https://sub.domain.com** (substitute with your server's IP address or whichever domain you've configured to point to *Audiobookshelf*) and you should be presented with the Initial Server Setup screen. Verify the location of your config and metadata, then create your root user. It goes without saying that it's a good idea to choose a different username from the 'root' presented, so change this and then set up a suitable password before clicking Submit.

You're now presented with the login screen – log in and you find yourself at the main screen where you're invited to add your first library. You're prompted to set the media type (books or podcasts) and give your library a suitably descriptive name. Each library can be assigned one of six icons, and then you set a default metadata provider, which provides the necessary information to flesh out the library with information such as a description, narrator and year. Google Books is the default provider, but there are alternatives: Open Library (**https://openlibrary.org**), iTunes and various regional Audible databases.

Finally, click Browse For Folder if you're running *Audiobookshelf* in *Docker,* or click inside New Folder Path to enter the path manually if you installed it through the repository. You can add multiple folders to the same library if you wish, then work your way through the Settings and Schedule tabs before you click Create.

Scan and explore

Click the *Audiobookshelf* logo in the top-left to return to the library's main screen. You're told it's empty and invited to scan it – before doing so, click Configure Scanner to tweak the scanner settings (for example, to choose an alternative cover image provider). Once done, click Scan Library and wait for the contents of the folder to be added to your audiobook library.

The initial scan should add a rich array of info – author, title, description, narrator and more – but if there are any missing elements, roll your mouse over a book's cover art and click the pencil icon in the top-right to open a Details pane with information in multiple tabs that you can view, amend and add to if required.

One shortcut to adding missing info is to switch to the Match tab to view current matches and try to find others using a different search engine or search terms. If you find the right match, click it, then choose what to include, make edits, then click Submit to use it.

The annotated screenshot reveals how to navigate the main web interface once you've got your first audiobooks imported, and now is also a good time to install the free mobile app on Android and iOS (you'll find links at **www.audiobookshelf.org**). It's incredibly easy to use, and closely mirrors the web interface. LXF

» GET AUDIOBOOKS

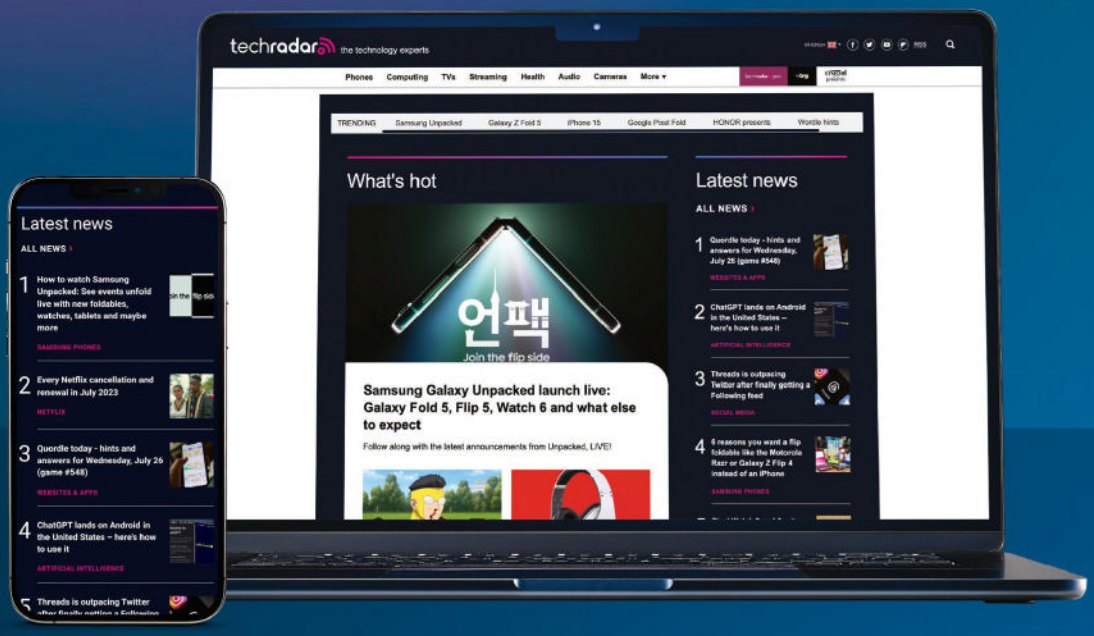
Wondering how to populate your audiobook server with great reads? You have three main sources. First, if you have an existing Audible collection, you need to download and convert your audiobooks into MP3 format using a tool like *OpenAudible* (£19.95, **https://openaudible.org**). Second, there are free sources of public domain audiobooks – mostly out-of-copyright classics – like LibriVox (**https://librivox.org**) and Project Gutenberg (**www.gutenberg.org**).

The final option is to rip your own purchased audiobook CDs. One word of warning – books are often chopped into lots of short tracks, which can result in lots of files. That's not a problem for *Audiobookshelf,* but if you'd prefer to keep things simple (such as one file per CD or chapter), you need a solution that stitches multiple tracks together, like *Fre:ac* (**https://freac.org**). To make this work, when ripping your disc, select the tracks you wish to combine and tick Encode To Single File before clicking the Encode button.

While you can manually tag your files if you wish, *Audiobookshelf* can do the hard work for you – including obtaining cover art. All you need to do is make sure your files are organised correctly, adhering to the following basic folder structure: Author > Series > Title > files

The Author folder supports various naming types, from First Last (such as Stephen King) to Last, First (King, Stephen), and Title folders can be named various ways, too – go to **www.audiobookshelf.org/docs** and select Directory Structure under Books for details.

Meet the technology experts



- The world's **most** comprehensive technology website
- An **unrivalled** mix of news, opinions, reviews and features
- **All-new design**, new homepage, new features and special reports
- Backed by **over 300 years** of editorial experience

techradar
the technology experts

www.techradar.com

Astro Pi Mission Zero blasts off into 2024!

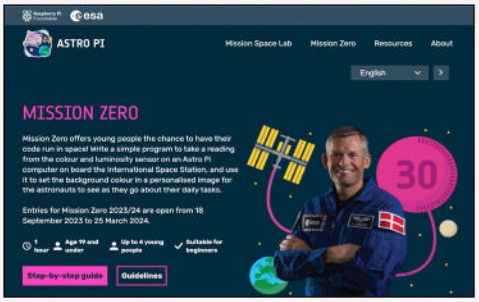
Go out of this world and enter the new competition for young coders that's now on!

We warned you before, but now it's here – the new Astro Pi Mission Zero for 2023/24 has blasted off! If you want to take part, you have until March 2024 to register and enter your coding project.

As a reminder, Astro Pi Mission Zero tasks young people with writing a simple Python program to take a reading using a sensor on one of the ISS Astro Pi computers and display a personalised pixel art image for the astronauts on board the ISS. This can be entered as an individual or team project.

This year's theme is 'flora and fauna', to help remind the ISS astronauts of the beauty of Earth's natural wonders. The overall aim is to display pixel art or animation on the Astro Pi's display and take a sensor reading. Every successful entry will be run for up to 30 seconds on the ISS and team members will get a unique certificate recognising their coding achievement.

The kind people at Astro Pi have provided a complete guide on how to get started and complete the Mission Zero coding project. They suggest it'll take around an hour to complete and just needs a web browser and internet connection, so there's no excuse not to visit **https://astro-pi.org/mission-zero/**.



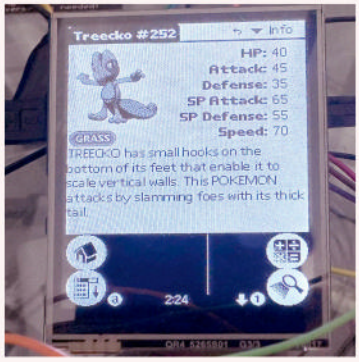
CREDIT: Raspberry Pi Foundation

Inspiring another generation of home coders!

Pi-alm0S PDA

Alive once more!

Palm PDA emulation has landed on the Pi Pico care of Dmitry Grinberg, who has shared an early demo of his platform, known as rePalm. Grinberg has created a custom kernel to run on Cortex-M processors. It meets all of the requirements for PalmOS, which enables the RP2040 to handle rePalm. Find out more: **http://dmitry.gr**



CREDIT: Dmitry Grinberg

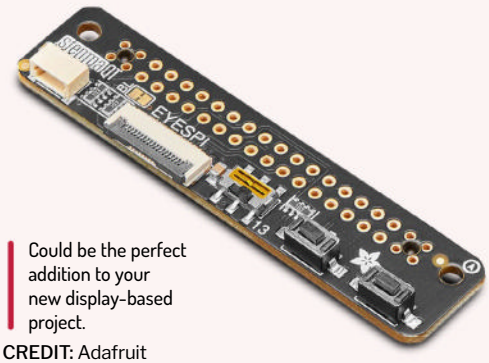
Relive those PalmOS PDA days.

The Pi Beret!

Adafruit EYESPI Pi.

The Pi has always had HATs, but what if you made a cute, thin HAT – would it be a beret? That's what Adafruit is betting on with its tiny GPIO HAT-like board. It provides an 18-pin SPI port, SDA/SCL I2C port and a couple of useful buttons and status lights, and costs as little as £5.

Find out more: **www.adafruit.com/product/5783**



Could be the perfect addition to your new display-based project.

CREDIT: Adafruit

Les Pounder works with groups such as the Raspberry Pi Foundation to help boost people's maker skills.

» RASPBERRY PI STILL RULES

Over the 11 years that the Raspberry Pi has been with us, there have been countless contenders for its crown: Orange Pi, Banana Pi, Libre Computer, Asus Tinkerboard, Khadas and LattePanda, to name but a few. These boards vary in price from as little as £10 to hundreds of pounds, and offer GPIO and faster CPUs, and promise better performance. Sure, some of these boards do perform better than the Raspberry Pi, but there is always a trade-off.

The Raspberry Pi 4 has a quad-core Arm CPU that now runs at 1.8GHz, and up to 8GB of RAM makes this a potent platform for projects. Boards such as the £230 Khadas Edge 2 Pro offer 16GB of DDR5 and an octa-core Arm CPU, but there is no GPIO. This little board is ideal for large display projects. The LattePanda Sigma retails for an eye watering £550 and packs a 12th-gen Intel CPU and up to 32GB of RAM. It also features an onboard Arduino Leonardo-based GPIO. With a powerful Intel CPU and Arduino microcontroller, the Sigma offers a powerful robotics platform. If your wallet can take it.

The Raspberry Pi can be seen as a good all-rounder. It provides a great GPIO, plenty of CPU power for machine learning and AI projects, and it maintains its original goal of making SBCs affordable. Before the Raspberry Pi came out, a typical SBC would retail for well over £100. Sure, prices have crept up, but they are still decent, and now that we have stock flowing once again, we can enjoy a few more slices of Raspberry Pi.

Raspberry Pi 5

The first new flagship in four years has **Les Pounder** and **Avram Piltch** fighting over who gets first dibs on the latest SBC.

IN BRIEF

The Pi 4 remains a solid choice; it doesn't need active cooling and has rock-solid support. If you don't need Linux, just the GPIO, then the £6 Pi Pico W has proven itself to be a versatile microcontroller. However, if you want the best single-board computer around, the Raspberry Pi 5 is your new leader.

After the longest wait between flagship Pi models ever (the Pi 4 was released in June 2019), the Raspberry Pi 5 is finally here and it's early! Eben Upton had said no new Pis in 2023, but here we are...

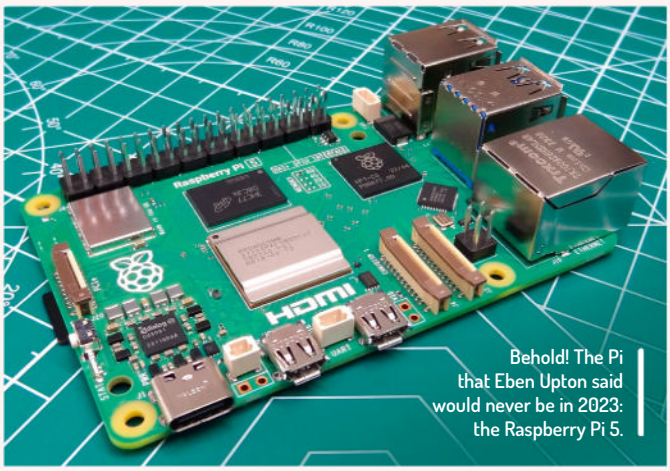
The Raspberry Pi 5 claims to have two to three times the processing power of the Raspberry Pi 4, already a powerful single-board computer. Available in 4GB and 8GB RAM capacities (with 1GB and 2GB to come), the Raspberry Pi 5 is the same basic size and shape as the Model 4 B, but adds a number of long-requested features, such as a built-in real-time clock, a PCIe 2.0 connector and a power button.

Perhaps more importantly, the Pi features a new quad-core 2.4GHz Cortex-A76 Arm CPU, a new Southbridge that promises to improve USB 3 throughput, and a new VideoCore VII GPU that operates at 800MHz. There's also a slew of little improvements, including a built-in fan header with mounting holes, faster and dual camera connectors, and a microSD card reader that works with higher-speed cards. All for just £4 more than the same RAM-spec Pi 4.

On the face of it, this is another classic Pi design, but look closer. Gone is the 3.5mm composite jack. The camera and display connectors are smaller, matching the Raspberry Pi Zero's 15-pin connector. These are now two connectors next to each other, and we can connect two cameras, two DSI displays or a mix of either.

The third interface is for PCIe devices. This is a PCIe 2.0 x1 interface for fast peripherals, and yes that means NVMe SSDs. We asked Upton about this and he confirmed that it will take all sizes of NVMe drives, but to do so, we need to use a specially designed M.2 HAT that was not ready for release at the time of this review.

The keen eyed among you will notice that the USB and Ethernet ports are swapped. It seems as though the Pi 5 is taking a cue from older boards. The Pi 4 saw the port positions swapped, and now they've been swapped back. So, you need a new case; this is possibly to ensure people don't cover the cooling fan using an old case.



Behold! The Pi that Eben Upton said would never be in 2023: the Raspberry Pi 5.

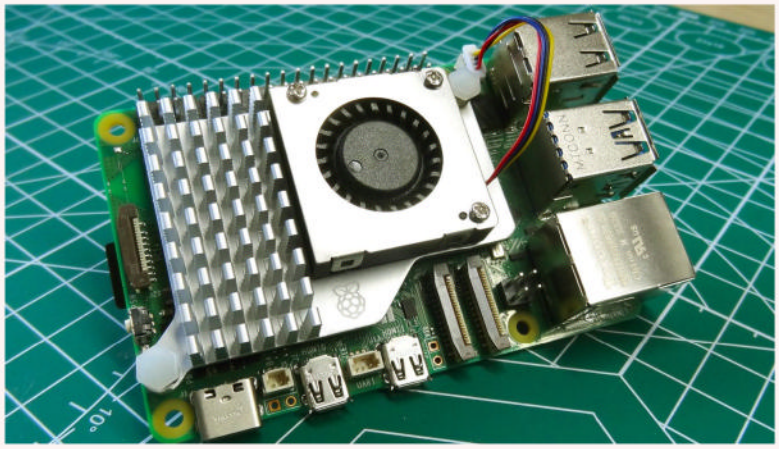
Let's address two new features for the Pi 5. Firstly, we have a real-time clock battery backup. Yes, your Pi can now keep the correct time without the need for an NTP server or an add-on board taking up space on the GPIO.

The other new feature is a power button. With the Raspberry Pi on, a single press brings up the shutdown/logout menu. Another press triggers a safe shutdown. This shutdown is more of a standby; another press of the power button starts the Pi 5 up. You can also program the OS to make the button do other things because it is a momentary button rather than a hard switch that cuts off electricity.

The Raspberry Pi 5 is the hottest of all the Pi boards we have ever used. At idle, without any added cooling, it sits at around 50.5°C and consumes around 2.7W. Pushing it with a stress test quickly makes the Pi 5 thermal throttle (which triggers at 82°C), dropping the CPU speed in an attempt to keep the CPU cool. Under stress, we hit 86.7°C (7W) and saw the CPU throttle from 2.4GHz down to 1.5GHz. Interestingly, we saw the CPU change speed a few times (2,311MHz, 2,256MHz, 2,201MHz and 2,146MHz), likely in an attempt to provide a balance between speed and cooling.

The question on your lips right now is: "How does this compare to the Pi 4?" At idle, the Raspberry Pi 4 runs at 45.7°C and consumes just 1.02W. Under stress, we see the temperature jump to 79.8°C and the Pi 4 consume 6.2W of power.

Initially at standby, the Raspberry Pi 5 consumes 1.3W, around half the power of an idle Raspberry Pi. The Raspberry Pi Engineering Team gave us instructions on how to lower the standby power and what a result we saw! Now our standby power usage is 0.05W, many times lower than before. This leaves the 5V GPIO pins at a high state, but the 3V3 pins are low. This could be a problem for your favourite HAT as it may not power off



The Raspberry Pi 5 needs active cooling and a new case.

along with the Raspberry Pi. A revision to the HAT specification, HAT+, will address this.

System performance

Opening apps is much faster. On the supplied microSD card, it took 5.5 seconds to open *GIMP* on the Raspberry Pi 5, compared to 10.8 seconds on a Pi 4. Opening *Firefox* took 5.1 seconds on the Pi 5 versus 8.6 seconds on the Pi 4. Boot times were around 18 seconds versus 38 seconds on the Pi 4 – with the same SD card and OS.

On *Sysbench*'s CPU test in single-threaded mode, the Pi 5 generated 2,729 events per second versus 1,766 for the Pi 4 (more events is better). When we upped the ante to four threads, the Pi 5 won again by 10,912 to 7,068 events, a 54% improvement.

When we ran the *7-Zip* compression benchmark, the Pi 5 delivered 9,543MIPS of compression versus 4,287 for the older model, a 122% boost. It also gave 13,231MIPs of decompression versus 7,568 for the Pi 4.

Streaming video remains the Achilles heel for the Pi, testing YouTube at 4K screen resolution playing just 1080p@60 video resulted in two-thirds of the frames being dropped, which is disappointing. Hopefully software updates will address this.

Providing the I/O Southbridge for the Pi 5 is the latest in-house silicon, the RP1. This provides more than double the USB bandwidth of previous models, with a dedicated four-lane 1.5Gb/s MIPI camera and display interface. This triples the total bandwidth for a combination of cameras and displays.

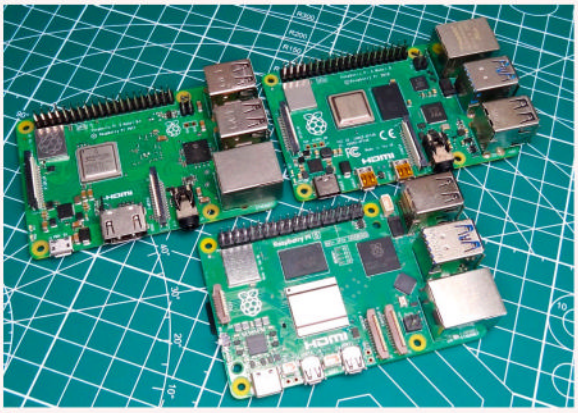
On *Sysbench*'s file IO test, the Pi 5 read at 12.75MB/s from a Kingston Canvas Go Plus microSD card and wrote at 8.5MB/s. Meanwhile, our SSD managed to read at 31.33MB/s and write at 20.89MB/s. On the Pi 4, the SD managed read at 8.78MB/s and write at 5.85MB/s. The SSD read at 12.96MB/s and had write speeds of 8.64MB/s. That's more than double the speed on both the USB 3.0 and microSD reader interfaces. Speaking of microSD, the Pi 5's card reader now supports higher-speed microSD cards that use the SDR104 standard.

Peripheral support

We performed our usual battery of GPIO Zero tests and had no issues. However RPi.GPIO proved to be an issue and this is down to some behind-the-scenes configuration. RPi.GPIO is most likely under the hood of your favourite Raspberry Pi HAT's software module and we failed to correctly test any of our usual third-party HATs. The official Raspberry Pi Sense HAT did test correctly, likely due to it using libgpiod over RPi.GPIO. Hopefully, most will be updated by the time of launch.

The Raspberry Pi 5 introduces multiple camera support to mainstream Pi boards. Compute Module users will be used to multiple camera support as it has been baked into the Compute Module IO boards since day one, but most Pi fans probably don't own a Compute Module. On the coding side, both libcamera and the Picamera2 Python module support multiple cameras and we successfully tested by passing the camera argument for libcamera (0 or 1) and with Picamera. LXF

Model	Raspberry Pi 5	Raspberry Pi 4
SoC	BCM2712 SoC Cortex-A76 64-bit 2.4GHz	BCM2711 SoC Cortex-A72 64-bit 1.8GHz
GPU	800MHz VideoCore VII, OpenGL ES 3.1, Vulkan 1.2	500MHz VideoCore VI OpenGL ES 3.1
Display	2x 4Kp60 HDMI HDR	2x 4Kp60 HDMI
RAM	1, 2, 4, 8GB LPDDR4X	1, 2, 4, 8GB LPDDR4
Storage	MicroSD (SDR104) M.2 NVMe SSD via M.2 HAT	MicroSD
GPIO	40-pin	40-pin
USB	2x USB 2, 2x USB 3 (5Gb/s)	2x USB 2, 2x USB 3
Connectors	2x 4-lane MIPI camera or display, PCIe 2.0 x1, UART, RTC clock power, fan power	2-lane MIPI DSI, 2-lane MIPI CSI, 4-pole audio and composite video
Networking	Gigabit Ethernet, PoE via PoE+ HAT	Gigabit Ethernet, PoE via PoE+ HAT
Wi-Fi/ Bluetooth	Dual-band 802.11ac, Bluetooth 5/BLE	Dual-band 802.11ac, Bluetooth 5/BLE
Power button	Soft power button	None
Power	5V 4A via USB C, PoE via Poe+ HAT, 5V via GPIO	5V 3A via USB C, PoE via Poe+ HAT, 5V via GPIO
Dimensions	85x56mm	85x56mm
Price	1GB TBC, 2GB TBC, 4GB £59.30, 8GB £79	1GB £35, 2GB £45, 4GB £55; 8GB £75



The Pi family is growing ever bigger. Here are the Pi 3, Pi 4 and Pi 5.

VERDICT

DEVELOPER: Raspberry Pi Foundation
WEB: www.raspberrypi.org
PRICE: £59.30 4GB (£79 8GB)

FEATURES	9/10	EASE OF USE	8/10
PERFORMANCE	8/10	VALUE	9/10

Significantly faster than its predecessor while costing almost the same price, its only (temporary) drawback is that some older HATs may not have support right away.

» Rating 8/10

STEAM LINK

Credit: https://github.com/ValveSoftware

Play Steam games on your Raspberry Pi

Les Pounder has just had his living room redecorated after a nasty flood – what better way to celebrate than by shooting aliens?

OUR EXPERT

Les Pounder is associate editor at Tom's Hardware and a freelance maker for hire. He blog about his adventures and projects at http://bigl.es.

YOU NEED

- **Pi 3B+ or Pi 4**
- **16GB SD card**
- **Network**
- **A PC running Steam**
- **A wired gamepad**

Linux gaming used to be a laughing stock. *Tux Racing* or *Neverputt* used to be the limits of our entertainment. But over the years, games have been ported and created for our favourite operating system. And then something marvellous happened. Valve's Steam Deck is a portable gaming machine powered by a custom AMD APU – but, more importantly, it runs a Linux OS. Using a mixture of Proton configurations, Steam OS can run many Windows games. Not every game is supported (we're looking at you, *Call of Duty)* but there are more and more being added to the ProtonDB site.

Playing Warhammer 40000: Boltgun on the lounge TV while sitting on a comfy sofa, thanks to the Raspberry Pi.

The problem with Steam Deck is that it is expensive – around £350 for the base model. But what if you had a spare Raspberry Pi laying around? *Steam Link* is an application that can turn a Raspberry Pi into a game-streaming device. All we need is a good network connection and a gaming PC on the same network. In essence, our Raspberry Pi is sending our inputs to the PC, which controls our game then streams the resulting video and audio feed to our Raspberry Pi, which is connected to a big TV in the sitting room. So, we can frag noobs from the comfort of our sofa.

In this tutorial, we are building a Raspberry Pi-powered *Steam Link* device and using a script to run the connection on boot, meaning we can plug in and start gaming with very little effort. The project can be built using a Raspberry Pi 3B+ at the absolute minimum. This model received an enhanced Ethernet port. While not quite Gigabit, it was faster than previous generations.

As a rule, *Steam Link* needs around 30Mbps bandwidth to stop the gaming going 'potato vision'. The best model for the project is the Raspberry Pi 4. We don't need the CPU horsepower nor the extra RAM it offers – instead, the Raspberry Pi 4's Gigabit Ethernet port is more than up to the task of keeping the lag down.

Getting started

Using *Raspberry Pi Imager,* flash the latest Raspberry Pi Buster Legacy OS release to a blank 16GB microSD card. Why Buster? It boils down to the latest Raspberry Pi OS release moving away from X11, something that *Steam Link* relies upon. Fear not, though, because the legacy OS option is available for some time.

For best results, make sure the Pi is connected via a Gigabit network to your gaming PC. You can also connect via 5GHz Wi-Fi but your mileage may vary. On your gaming PC, ensure that Steam is running.

Boot Raspberry Pi OS to the desktop and from there open a terminal, the icon for which is in the system tray. Now update the OS and software repositories ready for installing *Steam Link*. This could take a while, so grab a drink while it works:

```
$ sudo su
$ apt update && apt upgrade -y
```

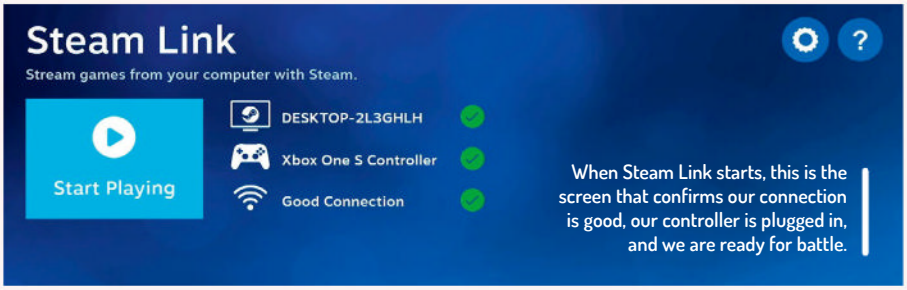
Install *Steam Link* via the terminal, then close the terminal when installation is complete:

```
$ apt install steamlink
```

Raspberry Pi 3B+ and Pi 4 1GB owners need to update the amount of video memory available. Go to the main menu, then Preferences > Raspberry Pi Configuration, and under the Performance tab, you

need to set the GPU memory to 128. Click OK and then reboot.

Click on the *Steam Link* desktop icon to launch the application. If the icon is missing, the app can be found under Gaming in the main menu. *Steam Link* starts updating and downloading extra software. There are times when you are prompted to interact, so answer the prompts to move the install onwards.



We now arrive at the *Steam Link* setup guide. Click on Get Started to begin. *Steam Link* scans your network for machines running Steam, and when found they are listed. Select the machine by left-clicking, then enter the authorisation number on your gaming PC and click Confirm. This triggers a network bandwidth test. *Steam Link* relies on a fast connection to stream games. Plug in a controller. Official Microsoft Xbox and Sony PlayStation controllers are supported. Click Start Playing to begin streaming your games.

The Steam user interface is designed to be simple and it defaults to opening our game library. From here we can launch any of our games and they stream over the network to our Raspberry Pi. When done, just click on the Xbox/PlayStation/Guide button and select Stop Streaming To Steam Link. To close the *Steam Link* launcher, press Esc.

Create an auto-booting Steam console

To launch *Steam Link*, we need to click on the icon each time we boot. Or do we?

Here is where we exercise a little Linux-Fu and create a games console that auto-starts. All we need to do is make sure our gaming PC is powered on.

On the Raspberry Pi, open the *Geany* text editor, found in the Programming menu. Use this to create a script with just two lines. The first line tells the script where to find the *Bash* interpreter:

```
#!/bin/bash
```

The second line calls the *Steam Link* application:

```
/usr/bin/steamlink %u
```

Save the file as **steam-console.sh** in the directory **/home/pi**. Next we need to make the file executable by opening a terminal and running this command:

```
$ chmod +x steam-console.sh
```

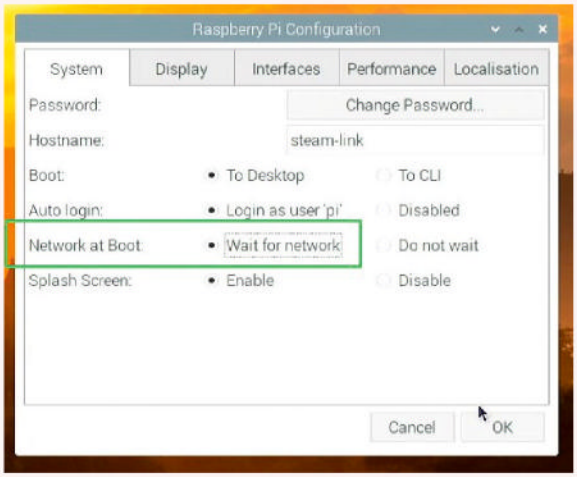
Keep the terminal open, as we next need to create a *Cron* entry that loads our script each time the Raspberry Pi boots to the desktop. Open *Crontab* in edit mode. If you are prompted to pick an editor, the decision is yours (we are not starting a best text editor war). We are sticking with *Nano* for this tutorial. Add the following as the very last line in the *Crontab* entry:

```
@reboot /home/pi/steam-console.sh
```

Now press Ctrl+X, then Y and Enter to save and exit. Reboot the Pi to test it boots directly into *Steam Link*.

Right now, you should have an auto-booting Steam console, but for the minority who don't (like us), we need to take an extra step. When *Steam Link* boots, it checks for any updates. But, when our Raspberry Pi boots, the network may not be ready, so this causes a `Couldn't download archive, aborting update` error message. We scratched our head for an hour and found that by going into the main Raspberry Pi menu, then Preferences > Raspberry Pi Configuration > System, setting the Network At Boot option to Wait For Network, then rebooting, the error was corrected.

We have now built a Raspberry Pi-powered *Steam Link* console. All we need to do is tidy up the wires, put the Raspberry Pi in a nice case, and then get down to saving the world! LXF



If you get a 'Couldn't download archive, aborting update' error when starting Steam Link, make sure that your Pi is waiting for the network.

» RUN STEAM OS ON OTHER MACHINES

Sadly, the Raspberry Pi can't run SteamOS, but you can run it on an x86 PC. Well, you can sort of run SteamOS. The installation image is only for Steam Decks, but some enterprising community members have made their own OS based upon it. HoloISO (**https://github.com/HoloISO/holoiso**) is basically SteamOS and provides the same experience, but on your own hardware. Paired with a decent CPU (Ryzen 5600X) and an AMD GPU, this OS can give you a better-than-Steam-Deck experience. Notice that we said an AMD GPU? HoloISO doesn't play well with Nvidia cards, largely because SteamOS is made with the custom AMD APU in mind.

Another alternative is ChimeraOS (**https://chimeraos.org**). This is very similar to HoloISO, but it refines the experience with a powerful web app to manage your games, controller-first interface and better support for controllers. The downside? It only supports AMD GPUs. If you have an AMD Radeon RX400 or better, you can enjoy some serious gaming. Where ChimeraOS really differs is support for other stores (Epic, GOG and so on) and for retro gaming emulators (though Steam Deck can run *EmuDeck* if you fancy retro gaming on the go).

So, it seems that Linux gaming is no longer a laughing stock.

A BIG BANG!

Matt Holder discovers what System76's Cosmic desktop and Pop!_OS distribution have to offer.

QUICK TIP

Pop can be downloaded from https://pop.system76.com/.

Apart from being a typographical nightmare, Pop!_ OS, which we'll refer to as Pop from now on, is a Linux distribution sponsored by System76, a US-based company that sells laptops, desktops, servers and peripherals. System76 specialises in providing the devices it sells with support for either Ubuntu or Pop, which come pre-installed. Selling devices with Linux is no simple task because drivers need to be written where they don't currently exist and configuration needs to be made for function keys and other built-in hardware.

So, let's have a look at Pop and see what it provides that isn't available from other distributions.

Pop is based on the most recent Ubuntu Long Term Support (LTS) release and its take on the desktop is provided by its own extensions and theming on top of Gnome. This turns out to have been an incredibly smart move because the team can build on the huge amount of security and other work that has been carried out by Canonical's Ubuntu engineers as well as the team working on Debian.

Pop differs from standard Gnome in that it uses a dock at the bottom of the screen and has a custom-made extension that allows the best of both worlds, with a tiling mode as well. The Pop installer also offers slightly different options from that of Ubuntu. The Pop software store gives access to the standard DEB files as well as the more modern Flatpaks.

Floating or tiling?

The debate between which desktop environment model is best is nearly as old (and as dangerous a question to ask in polite company) as which text editor is better. As in most situations, the answer is nuanced and different users weigh features in different ways. Floating window managers are the sort that you are most used to – every application opens a new window, leaving you to drag it to the size and location you want. Proponents of tiling window managers have a number of arguments, including that using the mouse is slow and bad for your wrists. The main difference between the two is that when opening a window on a tiling window manager, it's added to a grid with other windows. This means that all windows are in focus at once and there is no manual work required to lay them out. Keyboard shortcuts are available to select

between windows and move them around the grid. It is also easy to add applications into an exception list, so they can open outside of the grid – for example, your email client can open and fill the whole screen and when it is minimised, the grid then shows again.

To experiment with Pop's tiling functionality, navigate to the top bar, find the icon that looks like a grid pattern of three rectangles, and select the option to switch on tiling. Spend some time opening and closing apps. The next thing to understand is how to change the grid pattern. By selecting the top bar of an app, you can drag windows to different locations in the grid. Dragging one app on top of another allows them to be tabbed. Applications can also be resized by dragging the gap between two of them. This changes the size of other cells in the grid as well. Making this sort of switch is likely to take some time (maybe even weeks) to get used to the differences. The exception list can be opened using the same icon on the top bar and this also shows options to display shortcuts, too.

Cosmic future

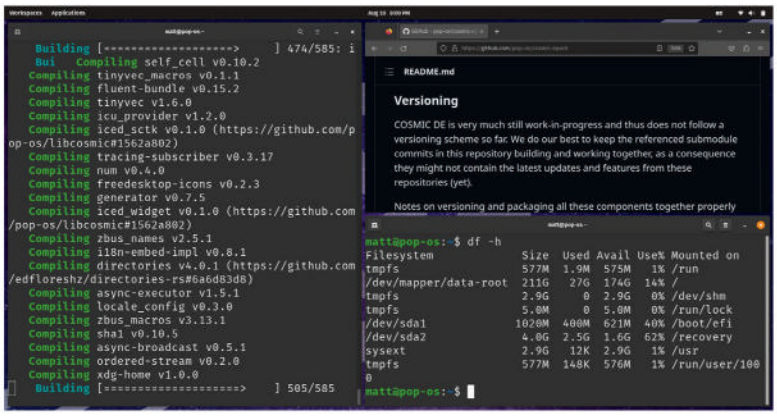
System76's own desktop environment is called Cosmic DE. The Pop experience is delivered using a number of Gnome extensions and custom apps, and it's clear that System76 wishes to take Pop in a very different direction from the way the Gnome team is going with regard to how it is changing or removing functionality that Pop relies on.

The Rust language is being favoured for Cosmic. It's relatively modern and was originally developed by a member of staff at Mozilla. Rust can be used for very low-level programs as well as higher-level ones. It has recently been adopted as a language that can be used for kernel development – only the second to have done so, alongside the venerable C. Rust guarantees memory safety by tracking memory usage throughout a program's lifecycle. System76's developers are also using the Iced library to provide the GUI components.

Cosmic DE is made from a number of components and is quite the undertaking. The team behind it is working on applets that can be used to customise your experience, an app launcher, a compositor, which is used to render output, as well as a notification system, software for on-screen display, a settings application, a library to render text, a text editor, themes, workspace support and libcosmic, the library that powers a lot of the aforementioned items. This is a non-exhaustive list and more information can be found at **https://github.com/pop-os/cosmic-epoch**.

While the desktop environment is very much a work in progress, the team does provide instructions on how it can be installed. Some of the items that are missing are settings applications, so config files need to be used and you should expect crashes! There are two ways to install Cosmic DE, with the first compiling the code on Ubuntu or another operating system. First of all, open a terminal and run the following commands:

```
$ sudo apt install just rustc libglvnd-dev libwayland-dev libseat-dev libxkbcommon-dev libinput-dev libgtk-4-1 udev dbus libdbus-1-dev libsystemd-dev libpulse-dev pop-launcher libexpat1-dev libfontconfig-dev libfreetype-dev lld cargo libgbm-dev libclang-dev libpipewire-0.3-dev -y
```



Tiling mode is easy to switch on and experiment with.

This step has installed all of the necessary libraries needed to compile the Rust code, which the new desktop environment is written in. Now enter:

```
$ git clone --recurse-submodules https://github.com/pop-os/cosmic-epoch
$ cd cosmic-epoch
$ just sysext
```

In this step, we are cloning the necessary Git repository, before changing into the newly downloaded source folder. In the final line, we are using the `just` command runner to compile all of the code that makes up Cosmic. It allows for 'recipes' to be written, which are used to run a series of commands. On our test laptop, compilation took around 15 hours, so be prepared for a long wait!

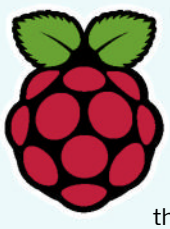
Once the `just` recipe has completed, you then need to move the newly created *Systemd* extension into the *Systemd* **extension** directory. Once copied, the extensions needs to be registered with *Systemd* before refreshing the extension:

```
$ sudo mkdir /var/lib/extensions
$ sudo cp -r cosmic-sysext /var/lib/extensions/
```

» RASPBERRY Pi SUPPORT

In December 2021, Jeremy Soller from System76 announced a preview of Pop for the Raspberry Pi 4.

To install it on your Pi 4, you need to write the image file to a microSD card. Alternatively, you can write it to a USB SSD and boot from that , but this is outside of this tutorial's scope.

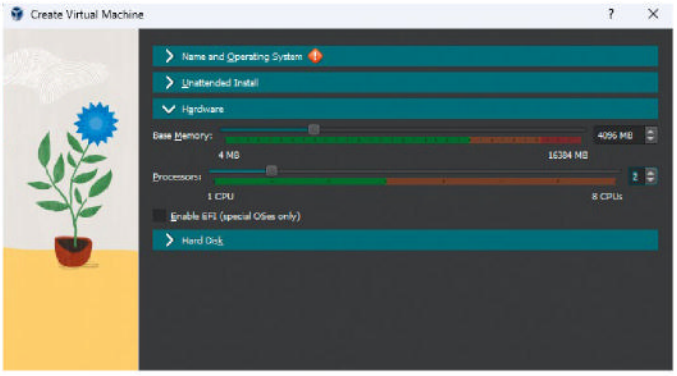


Raspberry Pi Imager is an excellent tool for preparing microSD cards, so download it from **www.raspberrypi.com/software/** and install using the downloaded file.

Now download the Pop image for the Raspberry Pi *(see Quick Tip, opposite)* and open the imager. From the left-hand panel, choose the option to select a custom image and navigate to where you have downloaded the Pop image. In the middle panel, select the microSD card that you've inserted into your laptop (make sure you select the correct device, so you don't wipe valuable data). Now, from the panel on the right, select the option to write the image to your card.

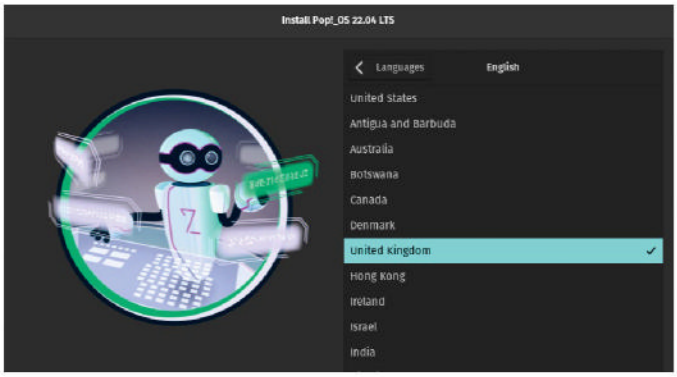
Once *Raspberry Pi Imager* has finished its work, disconnect the memory card and connect it back to the Pi. Connect your peripherals and power to your Pi. Once booted, you can configure things such as Wi-Fi and enjoy your new installation.

INSTALL AND CONFIGURE A NEW INSTALLATION



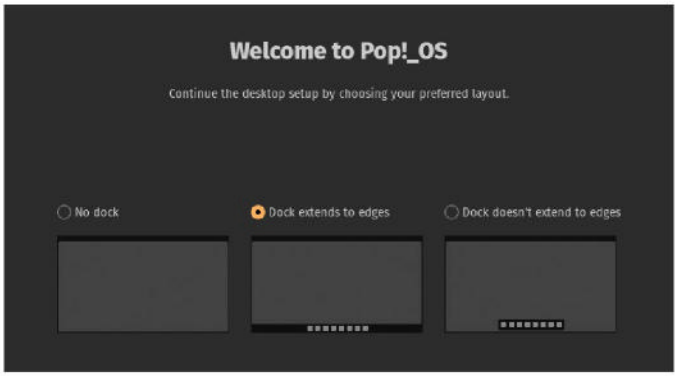
1 Create a VM and load the installer

Open *VirtualBox* or your virtualisation environment of choice and create a new virtual machine (VM). When prompted, pick a suitable name, then select the downloaded ISO file and set the basic hardware specification, which should be 25GB of disk space, 4GB of RAM and two vCPUs. Boot the VM and the installer should load.



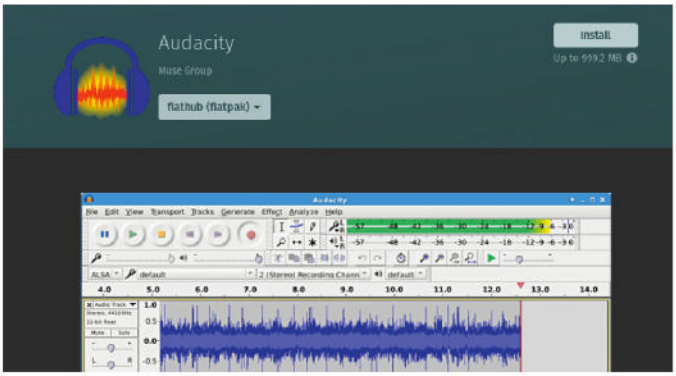
2 Install the OS

Once booted, the live environment starts and the installer loads. Use the wizard to set your time zone, keyboard language and any other localisation settings. During the installation, select your disk partitioning options and create a new user account. You are also asked if you wish to configure full-disk encryption. When completed, reboot into your new installation.



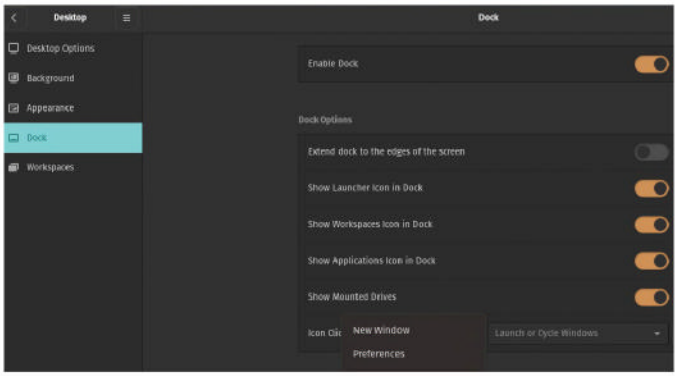
3 Complete the first-run wizard

Once you have rebooted your machine, if disk encryption was selected, you need to enter your decryption password. You now reach the login screen. When prompted, enter your username and password and when login has completed, the wizard loads. Follow through the steps to add any online accounts and set relevant options.



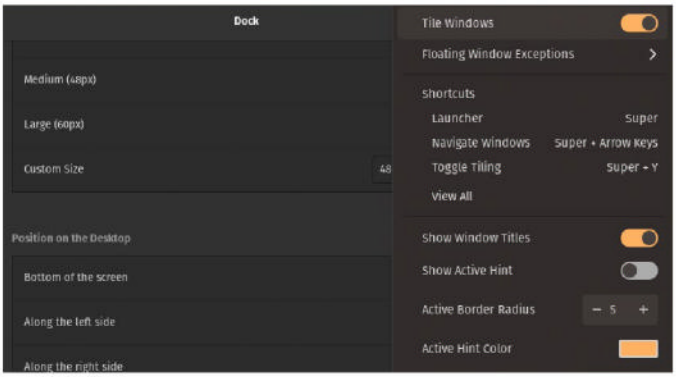
4 Adding software

By opening the *Pop!_Shop* app from the dock, you can install DEBs from the *Apt* repositories as well as Flatpaks – this provides you with an excellent selection of applications. Where the same application has options for both Flatpaks and DEBs, your choice can be made using the pull-down menu underneath the Install button.



5 Customise your dock

Once applications have been opened, they can be added to your dock by right-clicking on the icon and selecting the relevant option. Applications can also be removed from the dock by right-clicking and selecting the Remove option. You can explore the other options that the dock makes available by using the Settings option.



6 Experiment with tiling mode

Use the option on the right-hand side of the top bar to enable tiling mode. When selected, this automatically tiles windows on the screen, allowing multiple windows to be open at the same time. The menu also shows how to use keyboard shortcuts. Don't forget that applications can also be excluded from tiling mode.

```
$ sudo systemctl enable --now systemd-sysext
$ sudo systemd-sysext refresh
```

Next we need to enable Wayland at the login screen before installing some packages. Use your favourite text editor to edit **/etc/gdm3/custom.conf** and update the line to `WaylandEnable=true`. Wayland is a display manager that, while it cannot be described as new, it can be described as a lot newer than X, which has been used since the first GUIs were developed on Linux. Wayland is technically a standard, with multiple implementations available – for our purposes, however, Wayland can be considered as already described. It provides better support for tear-free video playback and different DPI values between displays.

Please note, your mileage may vary, and even though we went through these steps, we were not able to try out the new goodies.

There is an easier way to install Cosmic DE, however. This method involves first installing Pop, carrying out the previous step to enable Wayland at the login screen, and then installing the packages that the team has made available. Open a terminal and run the following command:

```
$ sudo apt install cosmic-*
```

Whichever method you have chosen to use, reboot, enter your username and password, and click on the option at the bottom of the screen to switch between Pop, Pop On Wayland and Cosmic.

Fun and games

The team at System76 is clearly very aware that some of its customers – and users of Pop on other devices – wish to use their graphics cards for gaming. When downloading the Pop image, you can select a specific version for Nvidia hardware, which comes pre-installed with the driver. The Steam client can be installed using the *Pop!_Shop* app or command line, and *Proton* can be used to increase compatibility and allow more Windows-only games to run on Pop. *Proton* is based on *Wine*, and the Valve engineers have put a huge amount of work into increasing game compatibility. Tools can also be installed to manage different versions of *Proton*, because some of the more bleeding-edge versions support more games. Documentation on this can be found at **https://support.system76.com/articles/linux-gaming/** and you may want to install *Lutris* as well, which supports installing games from multiple different sources.

5, 4, 3, 2, 1... POP!

As a brief aside, let's discuss System76's Launch keyboards, released for power users who wish to purchase a solidly-built mechanical keyboard. System76 offers three versions of the keyboard, which each contain a split spacebar as pair of keys. The keys are also backlit and there's a number of switch types to choose from at purchase time. While it is difficult to buy these keyboards in the UK, it is worth discussing them because the chassis design, firmware, PCB design and configuration software are all developed under an open source licence, so any information can be reused as long as the relevant licensing conditions are met. Using the *Configurator* software, custom key mappings can be set, such as using one Space key to fire a weapon when playing a game and the other to change your weapon. It is also possible to change the backlight colours and patterns.

The Cosmic desktop environment.

Over the last few years, System76 has invested heavily in its own manufacturing lines, which has led to the release of the Thelio high-spec PCs, with custom-made cases, Launch keyboard and now the Nebula case.

As can be seen from this article, what originally started as a rethemed Ubuntu installation with extra drivers has morphed into an incredibly polished, well-designed and well-thought-out product. With the attention to detail that the developers have given Pop, the distribution can be recommended for experienced and new users alike. The future looks very bright for Pop, especially when Cosmic DE has been released and thoroughly tested. We hope that you enjoy exploring Pop!_ OS. LXF

QUICK TIP

Learn more about Cosmic at **https://blog.system76.com/post/cosmic-de-tiling-redesign-and-libcosmic-rebasing/**.

» IMMUTABLE POP

We've discussed immutability in recent months, when covering Vanilla OS, NixOS, MicroOS and BlendOS. It basically means that the operating system partition is read-only. This is a security and stability consideration. It allows for updates to be applied in a very controlled manner and reduces the possibility of compromised software being installed, which could cause havoc on your installation.

Not much is known at the moment about Pop's immutable implementation, but it was positive to see a new repo being created on GitHub, where this feature can be worked on.

There's a good chance that the same sort of system will be developed as is used by other immutable distros.

Vanilla OS uses an AB partition system to apply updates to, so if any updates fail, the A partition is used. If updates are successful, the B partition is used instead. The B partition is then marked as the currently used one and then partition A is used to apply updates to.

MicroOS does not use the AB partition system, but updates are applied in a similar way nonetheless – they are staged and then applied at the next reboot. Having Flatpak support available is particularly important because packages are installed in the writable part of the filesystem and apps are run in a sandbox.

As has already been discussed, the System76 engineers are working hard on their own Cosmic desktop environment and hopefully immutability will be prioritised around this, too.

CREDIT: System76

GOFUL

Credit: https://github.com/anmitsu/goful

Fast file management

Shashank Sharma may be a gamophobe but he has no fear in admitting his commitment to the CLI thanks to the many wonderful tools on offer.

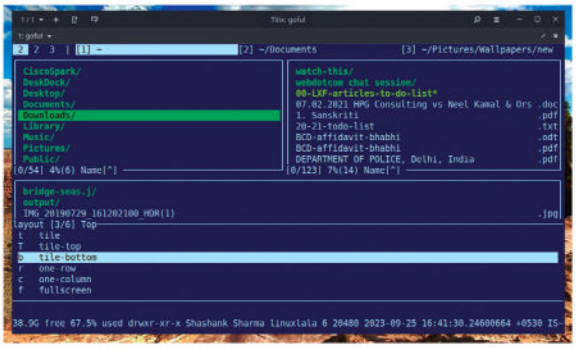
OUR EXPERT

Shashank Sharma is a trial lawyer in Delhi and an avid Arch user. He's been writing about open source software for 20 years and lawyering for 10.

CLI file managers have been around for decades. What makes some of the text-based filed managers stand out – from other CLI counterparts as well as the graphical alternatives – is their unique appearance or workflow. *Goful* is such a CLI tool, and if you're on the fence about switching to a CLI alternative from your current favourite, this might be the one that pushes you over the edge.

Written in Go and released under the MIT License, *Goful* is a dual-pane file manager that does the traditional CRUD tasks (create, read, update, delete) but also boasts features often missing from other CLI file managers, such as bulk rename, a powerful search filter, different layout styles, several colour schemes, the ability to work with archive files, and more.

Although *Goful* isn't in the software repositories of popular desktop distros, installation is straightforward. If you already have Go installed, you can run `go install github.com/anmitsu/goful@latest` to install *Goful*. If you don't already have Go installed, you can instead grab the latest tarball from the Releases section of the project's GitHub page (**https://github.com/anmitsu/goful/releases**). Extract the tarball and move the **goful** binary into a directory in your **$PATH**.



You'll appreciate the ability to tile the top and bottom of the interface if you frequently work with several directories.

You can now launch *Goful* with the `goful` command. The project doesn't have a man page, a help section or detailed documentation. This is a problem because the default interface has a lot of different elements, all navigable using keybindings. Thankfully, these are all mentioned on the project's homepage, but we'll also cover the essential ones in this tutorial.

Mastering the basics

Whether it's graphical apps or text-based utilities, arranging all the features into an easily navigable interface is always a challenge. You might find yourself feeling a little overwhelmed with all *Goful* presents you with by default.

To begin, the numbers on the top-left of the interface are the three workspaces on offer. Each one provides a two-pane interface by default. You can switch between the two panes in the current workspace using Tab. At the bottom of the interface, *Goful* provides helpful details about the disk space.

Before anything else, we recommend decluttering the interface by removing the details such as size and time from the two panes, and hiding hidden files. To do so, tap `v` to open the View menu at the bottom of the *Goful* interface. Use the up and down arrow keys to select Stat Menu and hit Enter. You can now choose to turn on or off the size, permissions and time. *Goful* refers to these as stats, which are displayed about all

» OTHER FEATURES

You can't always remember where you saved a certain file. Or you might want to search for files with a common extension. *Goful* has a quick solution to both problems.

To look for files or directories, press `f`, and type in the keywords you're looking for. *Goful* starts filtering results based on your entered keywords as you type. *Goful* performs a case-insensitive search if you enter lower-case letters, but it performs a case-sensitive search if you provide upper-case letters. Press Esc to exit the search, or use Backspace to remove characters from the search field. You can scroll through previous search history by pressing Alt+p or Alt+n.

If you're looking for files with a specific extension, you have to use regular expressions. To do so, you must hit `g` to enter the Glob search pattern. Hit Esc or Ctrl+g to exit the Glob pattern search. You can also perform a recursive Glob search by hitting `G`.

Goful also provides a shell mode that enables you to execute commands if needed. Press `;` to access the shell mode and type in the command you wish to execute. You can access the history of such commands by again accessing the shell mode and using the down arrow key to navigate through all the commands you've run in that mode.

files and folders. You can choose to toggle on only the ones you need, or toggle them all from View > Stats. The last option on the View menu is Toggle Show Hidden Files. By default, *Goful* shows hidden files.

In addition to the Stat menu, the View menu is home to the Layout and Look menus. Head into the Look menu to change the colour scheme, or to toggle the borders for the panes on and off.

The Layout menu can be used to switch the position of the panes. Experiment with the options on offer in the Look and Layout menus to find the appearance that best suits you. When navigating the View menu, you'll notice that each element has a defined keybinding. For instance, when you press `v` to access the View menu, you can see the following keybindings:

View menu	Function
s	Stat menu
l	Layout menu
L	Look menu
.	Toggle Show Hidden Files

You can access the different menus using the assigned keys, but these keys only work after you've already accessed the View menu. So, if you press `.` from the main *Goful* interface, it has no effect. You must first access the View menu by hitting `v` and then hit `.` to toggle Show Hidden Files. You must similarly access the Look or Layout menus before you can use the assigned keys for the different options on offer.

The files and directories are sorted by name by default. If you'd rather sort by time, size or extension, you can press `s` and select from one of the nine available options:

Sort menu	Function
n	Sort by name
N	Sort by name descending
s	Sort by size
S	Sort by size descending
t	Sort by time
T	Sort by time descending
e	Sort by extension
E	Sort by extension descending
.	Toggle priority

Filled to overflowing

When navigating the directories in the selected pane on *Goful*, you can use the up/down arrow keys, the *Vim* keybindings `j` and `k` or the mouse scroll wheel. *Goful* displays three elements of information at the bottom of each pane. Something like `[0/30] Top(1) Name[^]`.

Here, the first element shows the number of files selected out of the total files. Press Space to select files, and the first element changes. As you scroll down, the second element also changes, which show how many files are above the selected item. Finally, the third element shows the sort methodology used. This changes to `Size[$]` if you opt to sort by descending size for instance.

When working with directories, you can hit Enter to navigate into a directory and use the Backspace key to move to the parent directory.

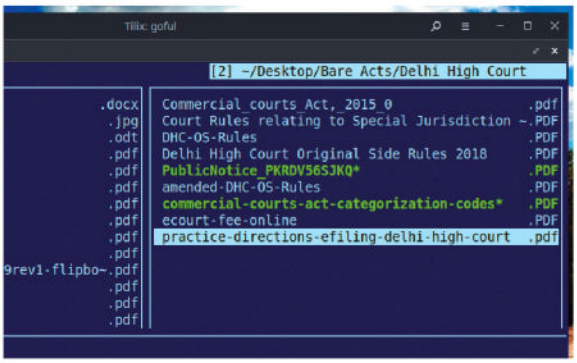
To copy files, first select them, then press `c`. *Goful* then asks you to specify the destination path. Hit Enter after you've specified the path and *Goful* provides a progress bar for the operation. You can use tab-completion when specifying the path to save time.

You can perform various operations such as rename, copy, move, delete, bulk rename, chmod and so on. For a list of these, press `x` to access the Command menu:

Command menu	Function
c	Copy
m	Move
D	Delete
k	Mkdir
n	New file
M	Chmod
r	Rename
R	Bulk rename
d	Chdir
g	Glob
G	Globdir

Instead of only two directories, you can also open additional directories in the current workspace by pressing Ctrl+o. To close a directory, select it using the Tab key and then press Ctrl+w. You can similarly switch workspaces by pressing Alt+f and Alt+b. Press Alt+Ctrl+o to create additional workspaces and Alt+Ctrl+w to close a workspace. All settings, such as layout, sort and so on are specific to the current workspace only, so you have to set them again for each workspace.

We've worked with a host of file managers over the years, including many text-based solutions. While each has its own pros and cons, more than anything else *Goful* has a sense of style about its operation that we find sleek and beautiful. But if you're still not convinced, refer to the Other Features box *(opposite)* for a quick rundown of some of its additional capabilities. LXF



Files with executable permission are shown with a star next to their name and sport a different colour.

QUICK TIP

As with the View menu, the assigned keys for the Sort and Command menus can only be used after you've accessed the primary menu. For instance, to create a new directory, you must press `x` to open the Command menu and then press `k`.

FOTOXX

Credit: https://kornelix.net

Take command of your photo collection

Nick Peers reveals how to organise your photos, plus improve, repair and enhance them – all with a single tool.

OUR EXPERT

Nick Peers may fancy himself as a wannabe photographer, but the sad truth is that without his vintage Panasonic bridge camera's intelligent auto setting, he'd be screwed.

Are you on the lookout for a tool to organise, edit and manipulate your photos? *Fotoxx* is a powerful free open source tool developed with one eye on semi-professional photographers, but you don't need to be an enthusiast with an expensive digital camera to benefit from its powerful feature set.

In this tutorial, we're going to introduce you to the full range of *Fotoxx's* capabilities, from indexing and organising your photos to editing all or parts of the image for a range of purposes, whether fixing common mistakes or looking for ways to make your photos pop.

Installation

Fotoxx exists in most distros' universal packages, but it's probably out of date. The current build at time of writing is 23.60; go to **https://kornelix.net/downloads/downloads.html** and click the **fotoxx-23.60-bin.tar.gz** link, saving the file to your **Downloads** folder. From here, install it by issuing these terminal commands:

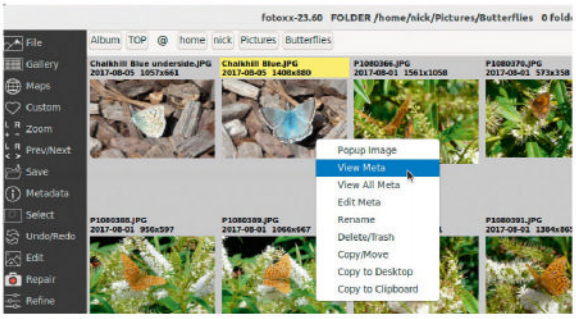
```
$ cd Downloads
$ tar -xzf fotoxx-23.60-bin.tar.gz
$ cd fotoxx && chmod u+x install.sh
$ ./install.sh
```

Log off and on again, and the program shortcut should appear in the launcher. Click this and you'll see an Outboard Programs window appear, listing all required (and recommended) programs. If necessary, make a note of any packages listed under 'Install following programs…' and install the package name via the terminal (for example, `sudo apt install dcraw`). Also install any packages required for optional file types you plan to use, such as HEIC and WEBP files. Once done, relaunch *Fotoxx,* which should now function correctly.

First run

You'll see the First Startup wizard pop up, inviting you to index your images so they're added to *Fotoxx's* internal database. While you can defer this, it's best to leave the default Index Image Files Now setting enabled and click Proceed. You're prompted to select a minimum of one top index folder. By default, your entire **home** folder is selected, but if you wish to change this (or add folders), click Select to do so.

Now is also a good time to choose what metadata *Fotoxx* will include in its index to allow you to quickly search by the selected tags. Click Select next to Extra Metadata Items To Include In Index followed by Yes, then browse the list, clicking on any items you wish to include in the index, or click Other Item to manually define tags not in the list. Click OK when done.



Fotoxx's Gallery view can be used to browse your photos by different criteria. Right-click a photo for a small selection of options.

Click Proceed, and a build index window opens to keep tabs on the process. The speed depends on your PC's hardware, the speed of the drive containing the photos (NVMe or SATA is quicker than traditional HDDs), and how many photos you need to index.

Once the process is complete, the main *Fotoxx* window shows a series of folder icons – click the OK button to remove the build index window and you're ready to start using the program proper.

Navigation made easy

The folder view provides you with a familiar(ish) way of navigating your library. Each folder displays its name followed by the number of subfolders inside it, along with the number of images that are in the folder itself. Click a folder to browse inside, where you'll see the view update. If there are pictures inside the folder, scroll down below the subfolders to view thumbnails.

Each thumbnail is accompanied by info about the photo – when it was taken, its filename, file modified date, resolution and file size. Click an image to select it for viewing or editing, or right-click it for more options, from viewing (and editing) metadata to file operations (rename, delete, copy/move, copy to clipboard).

When viewing an image, you can quickly go through the other photos in the folder using the left and right

QUICK TIP

One of *Fotoxx's* strengths is that it supports the RAW file format, not just for viewing and editing, but also for converting to other formats. There's even a Batch RAW tool for converting multiple files on the fly.

arrow keys, or click Gallery to return to the previous view. When browsing in Gallery view, change the size of the thumbnails being displayed by rolling your mouse over the Zoom button on the left, then left-clicking to make the thumbnails larger, or right-clicking to shrink them. The same tool can be used to zoom into and out of any images you're editing.

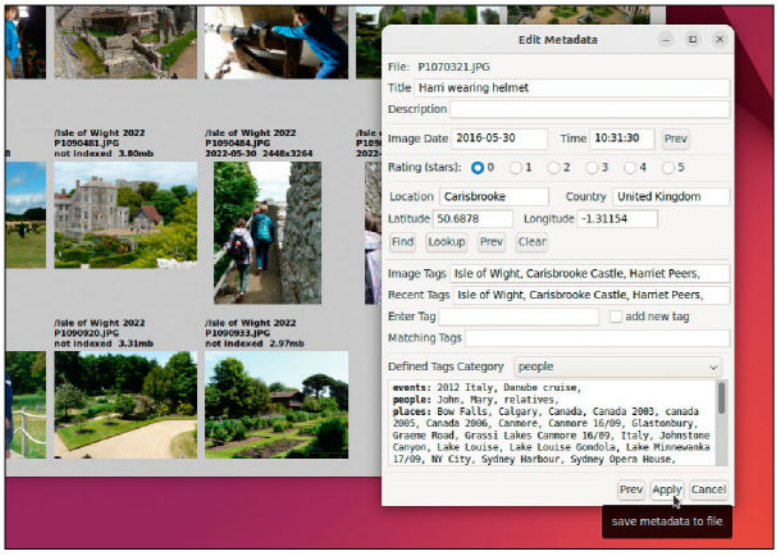
Also check the toolbar at the top of the program window – here you can see a folder breadcrumb enabling you to quickly navigate up the folder tree. To the left of the @ symbol are two buttons: click Top to jump between common locations via a drop-down menu; click Folder Tree to manually browse to a specific folder within your library. Clicking Album tells you no albums have been found – see the box *(below)* for more information on placing selected images in one or more albums.

Find files quickly

Browsing files is all well and good, but if you have a large library, you'll want to know how to quickly find a specific image. This can be done by opening the Metadata menu by clicking the 'i' button on the left. There's a comprehensive search tool that enables you to build complex search queries across tags, keywords, dates, places, ratings and more (or simply enter some keywords into the Search Files field and click Proceed to see what comes up), plus two handy summaries.

The first is Places/Dates, which enables you to group your images by country, country/location, country/location/date or date/country/location. By default, dates are combined into single days only, but you can extend this however you wish (such as seven for weeks, or 30 for months). This view is somewhat limited because it ignores any geo-coordinate data you may have entered, relying instead on what's recorded in the photo's Location and Country tags.

The second option is to choose Timeline > All Images, which distributes your photos according to year and month – simply click an entry to filter the gallery so only photos from the selected month or year are shown.



Take the time to add the correct metadata and you open up new ways of exploring and organising photos.

Refine your view

The Gallery can be configured to show exactly what you want it to by clicking its button on the left to reveal a drop-down menu. You start off with the default Thumb View, but if you want to display more information about each photo, you should choose Meta View instead.

Beneath this are two filters that can help you to quickly locate items of interest: Recent Files provides a breadcrumb trail of images you've recently viewed, while Newest Files enables you to choose to display the newest files based on their internal photo date or file modification date.

A Gallery Sort option enables you to change the order in which files are displayed, while Gallery Screen

CREATE ALBUMS

Fotoxx enables you to group photos into albums, which work like they do with their paper equivalent – you collate a series of snaps based around whatever criteria you choose, then arrange them in a desired order. Think of them as an alternative way to organise your photos without having to manually move or copy them into a single folder – in fact, they're a text file.

Everything you need to get started with the feature can be found by clicking Gallery and choosing Manage Albums. Start by clicking Create Album to name your album and decide whether to create an empty album or populate it with the current file selection, current gallery view, or by manually selecting files in the usual manner. Once done, click OK.

Going forward, access the album's contents by clicking Album at the top of the screen and selecting your album title. Rearrange items simply by dragging and dropping the files into the desired order, or right-click and choose Remove From Album to remove the currently selected photo (the photo itself isn't deleted). You can even select a group of new files to add, then insert them at a specific point in the album by right-clicking an existing photo and choosing Add Selected Files Here.

The Gallery menu offers two more album tools: Update Album enables you to replace one file with another (typically a newer version), while Album Mass Update provides even more ways to process new and changed files.

Albums collate a selection of photos any which way you like. Individual photos can reside in more than one album.

QUICK TIP

Frequently forget where favourite tools are found behind one of *Fotoxx's* many buttons? Click Custom and choose Edit Menu to see how to build your own hand-picked menu using a text config file.

enables you to filter the files shown in the current view using keywords (including or excluding).

If the Places/Dates filter disappoints you, the good news is that *Fotoxx* offers a Maps view. Click this and any photos with geo-coordinates attached to them can be pinpointed on a map – just zoom in to where the red dots are, then click one to see which photos have been tagged at that location.

Edit metadata

As we've seen so far, *Fotoxx* works best with images packed full of quality metadata – where the photo was taken, when it was taken, who's in it, and so on and so forth. Thankfully, not only can you view a photo's metadata in *Fotoxx*, but you can edit it too – simply right-click and choose Edit > Metadata to do so.

You'll see fields for adding a title and description, attaching a date, time and place, rating the photo out of five stars, and including user-defined tags. Most are self-explanatory – after filling in the Location and Country fields, click Lookup to quickly populate the geo-coordinate fields if you don't know exactly where a photo was taken (if you do, take the time to paste in the correct latitude and longitude coordinates using an online service such as OpenStreetMap (**www.openstreetmap.org**), which ensures photos can be tagged to an exact building or location.

One way to bring order to your photo collection is through keyword-based tags. To speed things up (and keep things consistent), *Fotoxx* records all existing tags and a complete list is shown beneath the Defined Tags Category drop-down. If you've been diligent, they may even show up in logical categories – events, people, places or things – but it's likely they're filed under Nocatg. No matter – to add one of these to the current file, just click it from the list, or enter your own choice of tag into the Enter Tag field and tick Add New Tag.

As an aside, if you would like to organise your tags, open the Metadata menu and choose Manage Tags. Click a tag in the list, then manually enter the category

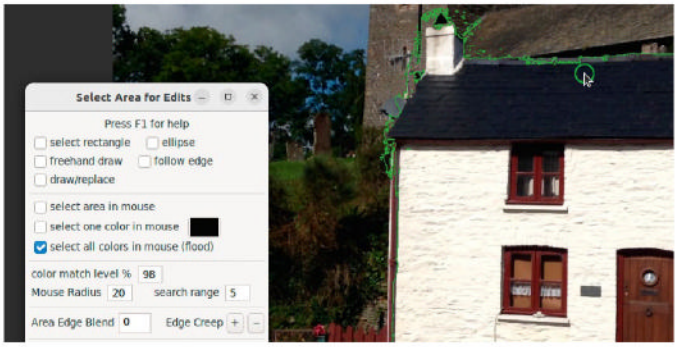
USE THE FLOOD SELECTION TOOL

1 Flood selection

One of the easiest ways to select a complex object is with the flood selection tool. Start by selecting your object and clicking Select > Select Area. When the dialog opens, tick Select All Colours In Mouse (Flood). The cursor changes to a small circle. Place this in the centre of the object you wish to select, then click and drag with the left mouse button.

2 Select your object

You'll see the selection start to expand as you click and drag around inside the object. Hopefully it'll detect the edge of the object and create a neat selection, but it's likely the selection will expand outside the object. Ignore this for now and keep selecting everything inside the object so there are no missing bits like those shown above.



3 Shrink the selection

To remove unwanted parts of the selection, move the cursor outside the current selection, then hold the right mouse button as you click and drag inwards towards it – you should see the selection reduce in size. If it reduces too far, try upping the Color Match Level % figure to 95 or even 98% before left-clicking to expand the selection again.

4 More adjustments

Keep adding and removing parts from the image – it can take some time. If certain areas are proving particularly fiddly, look at reducing the size of the mouse cursor using the Mouse Radius setting. Also experiment with reducing the Search Range figure to 2 or 3 for more precise adjustments. When you're happy with your selection, click Finish followed by Keep.

you'd like to place it in before clicking Create to move it to that category.

Edit your photos

Fotoxx isn't just a tool for collating and organising your photos, it's a capable image editor in its own right. Unsurprisingly, these tools are geared towards photos, and are either accessible from *Fotoxx's* left-hand pane or by right-clicking an image you've clicked on.

Fotoxx divides its editing tools into six categories – and a handful of commonly used options from the first three (Edit, Repair and Refine) are replicated on the right-click menu. These may be basic, but *Fotoxx* doesn't skimp on offering helpful touches such as mirroring images (Rotate) or cropping to a fixed or preset aspect ratio (Crop).

Notable tools worth exploring include Retouch for adjusting brightness, contrast and colour manually. You have a choice of using a clickable levels graph or simple sliders to make adjustments, complete with options for selecting black levels and white balance. You can even save favourite settings as a preset for use across multiple photos.

You'll see a handy Undo/Redo button for correcting editing mistakes, and don't forget to click Save when you're happy with your changes – when prompted, you can choose to create a new file, change the file type or overwrite the existing one.

Editing made easy

These manual tools are all well and good, but *Fotoxx* also offers a wide range of semi-automatic tools for editing, repairing, refining and applying effects to your images. Let's start by examining what you can do with a photo blighted by poor exposure or suffering from a lack of detail (say through haze or fog).

Open the affected image, click the Refine button and choose Flatten Dist to open the Flatten Distribution dialog. You'll see a series of sliders – five individual zones from dark to light, or a catch-all All Zones slider. We recommend starting with the latter to make broad changes, then fine-tune individual sliders to see how they affect things. You'll be surprised at just how much detail there is lurking in the shadows and behind the fog, waiting for you to bring it out.

The Refine menu also offers a dedicated Defog tool, Local Contrast, Gradients (low brightness), Global and Zonal Retinex (restore colours), while the Repair menu is home to no fewer than four different sharpen models for countering the effect of unwanted blur. There are also seven models designed to reduce the noise in photos shot in low light, and a Smart Erase tool – select the area you want to delete, and *Fotoxx* attempts to fill it with detail from the surrounding area so the unwanted object is no longer there.

Select part of your image

Sometimes you only want to edit part of your image, or perhaps you'd like to cut out part of it for use in a separate image. *Fotoxx* offers an array of selection tools you can use, from the familiar geometric shapes (rectangle and ellipse) to more precise tools such as Follow Edge, where you carefully trace a line around your chosen object or part of the image.

Another effective tool is flood selection *(see step-by-step guide opposite).* This sweeps up part of the image as you click and drag the mouse, and while it works best on clearly defined objects, it can be refined to make quite precise selections even when the background and foreground aren't that different.

When working on your selections, you can switch between tools without affecting your current selection. When clicking Finish, you're prompted to visually confirm your selection – right-click outside your selected area to verify it's correct, then click Keep; otherwise close the dialog and continue working.

Even after you've confirmed your selection, you can still return to it again – click Select > Select Area again to make changes, or click Clear if you're done with the selection and want to start again from scratch.

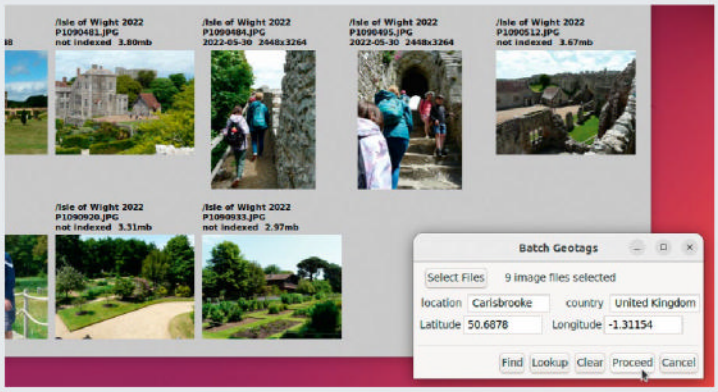
The Select menu offers lots of options for working with your selection – from inverting it (so edits only affect non-selected portions) to saving the object as an individual image (with transparency for pasting into other images). Press F1 for a detailed guide to what all the options – including other selection tools – do. LXF

» BATCH PROCESSING

Click Batch and you reveal a host of useful tools for working on your photos at the file and metadata level. There are options to convert images between formats, rotate (Batch Upright), rename and move/copy them, plus a range of metadata-related options from applying tags in bulk to geotagging a group of files taken in a single location.

Start by selecting your tool of choice from the Batch menu, then click Select Files. This works in the same way as the Select Files option under Gallery: a new dialog opens and clicking any file in the Gallery view adds it to the list, which you can then edit using the Remove/Clear buttons as required. Once done, click OK.

From here, you then need to fill in any required information – such as location and country tags, plus geo-coordinates for geotags – or choose your options, such as which file type to convert your selected files to. When you're happy with your choices, click Proceed and *Fotoxx* starts ploughing through the list, one file at a time. Depending on the action and number of files you've selected, this can take a while.



Quickly apply consistent metadata across multiple files using Fotoxx's batch processing tools. Geotags support location and geo-coordinates.

QUICK TIP

Find out more about *Fotoxx* at its homepage (https://kornelix.net/fotoxx/fotoxx.html), where you'll find a link to a YouTube video offering a basic overview of *Fotoxx* that lasts just under an hour in total.

WORDPRESS

Credit: http://wordpress.org

Part One! Don't miss next issue, subscribe on page 16!

Twenty years of WordPress websites!

In the first part of this new series, **Michael Reed** takes a look at what the content management system WordPress can serve up.

OUR EXPERT

Michael Reed started with HTML website design while skipping between free ISP trials and using Mandrake Linux back in the '90s. These days, he's more of a WordPress and Ubuntu man.

WordPress is a CMS (content management system), and it's 20 years old this year. It started life as a simple blogging platform but, over the years, its versatility has been greatly expanded by the developers. As well as its central purpose as a blogging tool, it can host a huge variety of apps and plugins. These mean you can use *WordPress* to set up forums, galleries and, well, pretty much anything that involves making content available online.

Quite often, when people have an online project in mind, they use *WordPress* as the starting point to organise collaborative facilities. There is some flexibility as to how you install *WordPress*, and once installed, you can customise how it looks and works. You can then add plugins that greatly extend the functionality with things such as forums, to-do lists and calendars that help pull together an online community. We'll cover such projects over the course of the series.

Local installation

We'll start with the traditional method of hosting *WordPress* on your own server using the Apache web server. It's a good place to start because it will familiarise you with how *WordPress* hosting works. These examples assume you are running a Debian-derived Linux distro such as Ubuntu, but the overall procedures are the same for any normal Linux distro, even if the odd package name is slightly different.

WordPress hosting doesn't necessarily need a powerful computer. The actual amount of required processing power or RAM depends on how many clients you have to serve simultaneously. If you only expect to have a few people accessing the site at once, any computer with around 4GB of RAM should be sufficient. This includes a Raspberry Pi. If your traffic expectations are modest, there's nothing wrong with installing the server on to your main PC or a virtual machine running on a virtualiser such as *VirtualBox*.

You don't even need physical access to the VM or computer running the server, as you can SSH into the server for the first part of the tutorial, then access the *WordPress* install via a web browser for configuration and management. See your distro's instructions if you're unsure about how to set up an SSH server.

These days, WordPress defaults to a fairly minimalist look that makes a lot of use of white space.

How it works

WordPress is hosted on a LAMP (Linux, Apache, *MySQL*, PHP). Linux is the best platform (of course!) for hosting things, and Apache is the web server. When browsing the web, web browsers connect to a web server that is located at a given IP address or a domain. When the web server receives a request from a web browser, it serves the requested content in a form that the browser can understand.

WordPress provides what is called dynamic content. That means the individual pages are built on the fly when they are requested, rather than being stored in a complete form. For example, a typical *WordPress* page might contain elements such as a sidebar, a header, some graphics and some text. When a request is made, *WordPress* fetches content from its database (a *MySQL* database in this example) and serves it up. As it does this, it applies the selected theme, which affects the layout and appearance of the content, and it can serve different information to different types of user, such as those who are logged in or visitors who are not.

WordPress itself is written in the PHP programming language, and that's something that Apache is designed to handle. For that reason, PHP is a requirement that can't be swapped out for anything else. There are, however, some variations you can make with the setup. Although we're using Apache because it's the most popular and well-supported web server, Nginx is also fairly popular and potentially offers higher performance. It's worth considering if you're anticipating high traffic loads. We're using the *MySQL*

QUICK TIP

As soon as you start working with this tutorial, open a plain text file to make notes about things such as the usernames, passwords, folder names and IP addresses you're going to rack up.

database but other databases are available and work with *WordPress*. Linux is our platform of choice, although other platforms can host *WordPress*.

Install the prerequisites

Start by installing the Apache web server with `sudo apt install apache2`. You can test that this has worked by browsing to **localhost** in your web browser of choice, and you should see Apache's welcome page. This page is located in Apache's root folder, and it's called **index.html**. In a terminal, try navigating to this directory and finding this file, as the location varies with different distributions of Apache and you need to know Apache's root HTML directory later on. It's typically **/var/www/html** or something around that region. Anything you put in here can be accessed by a browser. For instance, if you placed a file called **hello.txt** into that directory, you could display it in a browser by navigating to **http://<your domain or IP address>/hello.txt**.

By default, Apache closes down properly when the machine is shut down and automatically starts up when the computer is booted up. As Apache is installed at the system level, you do not have to log in to the desktop to start the server.

Install the PHP prerequisites:

```
sudo apt install php libapache2-mod-php php-mysql
```

As well as installing PHP itself, these packages configure Apache so that it can run PHP code and connect to the *MySQL* database.

Install and configure MySQL

Install the *MySQL* server with the following command:

```
$ sudo apt install mysql-server
```

Log into the *MySQL* server with `sudo mysql -u root`. This puts us in the command-line interface for *MySQL*, and it should look like this:

```
mysql>
```

Now we are logged in to *MySQL*, it is time to create the database that *WordPress* will use to store its content. Use `CREATE DATABASE wordpress_db;` to create a database called **wordpress_db**. *MySQL* commands tend to be upper case to aid clarity, even though they work if they are lower case. However, the semicolons at the end of a command are needed.

We have to create a user that *WordPress* can use to access this database, and we'll call it **wp_user**. In this example, the password is **wp_pass**, but you can customise this yourself: `CREATE USER wp_user@localhost IDENTIFIED BY 'wp_pass';`.

We grant that user the appropriate privileges when accessing the **wordpress_db;** database with:

```
Your MySQL connection id is 8
Server version: 8.0.34-0ubuntu0.23.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE wordpress_db;
Query OK, 1 row affected (0.59 sec)

mysql> CREATE USER wp_user@localhost IDENTIFIED BY 'wp_pass';
Query OK, 0 rows affected (0.69 sec)

mysql> GRANT ALL PRIVILEGES ON wordpress_db.* TO wp_user@localhost;
Query OK, 0 rows affected (0.52 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.15 sec)
```

Setting up the MySQL database. It's just a few commands to set up the database and a user, and to grant permissions.


```
GRANT ALL PRIVILEGES ON wordpress_db.* TO wp_user@localhost;
FLUSH PRIVILEGES;
```

Exit the *MySQL* command line with `exit`.

Adding the database credentials. Some note-taking and cut and paste are a good idea here.

Install WordPress

Go to the *WordPress* website (**https://wordpress.org** rather than **https://wordpress.com**) and download the archive by following the Get WordPress link. This is usually a ZIP file called **latest.zip**, and we downloaded it into our **home** directory at the command line with `wget https://wordpress.org/latest.zip`. Unzip the file (`unzip latest.zip`). From here, you have to make the decision of where to place *WordPress*. If you delete the **index.html** in Apache's root folder and move the *WordPress* files there, users can access your *WordPress* site with a URL like **http://<your domain or IP address>**. If you install *WordPress* into a subdirectory, that is reflected in the URL, giving an address such as **http://<your domain or IP address>/wordpress/**. We're going to assume the latter rather than the former arrangement.

Move the entire **WordPress** folder that was created when you unzipped the archive: `sudo mv wordpress /var/www/html/`. Set the ownership of that folder and

QUICK TIP

Wherever we have specified 'localhost' as part of the URL of the server, you can use the IP address of the server if you are accessing it from another machine on your network.

» OTHER INSTALLATION METHODS

We've outlined a fairly classical approach to installing *WordPress*, and this gives you a lot of options for how you set it up. However, there are plenty of other ways of setting up *WordPress*; although, they're not necessarily any easier than a manual installation.

Using the official *Docker* images is an excellent way of installing and maintaining a *WordPress* site, but you need to know your way around *Docker*. You need to know how to add another container for the database and join it to the *WordPress* container on an internal *Docker* network, and how to handle *Docker* volumes. Any modifications also require logging into a *Docker* container.

Distros such as Ubuntu include *WordPress* in the repos, but we weren't that impressed as it led to a non-standard file layout, it still needed some setting up at the command line, and it was out of date.

Most professional web hosting includes the PHP and *MySQL* that *WordPress* needs as standard. You normally set up the database through the hosting company's interface. Most hosting companies have an option to automatically install the latest *WordPress* into your hosting space for you, and this might be the most convenient option of all, at the cost of some configuration freedom.

QUICK TIP

The 'root' *MySQL* user is for administration and isn't the user that *WordPress* will use to access the database, nor is it the root user of your Linux system.

its content so that Apache can access it with:

```
$ sudo chown -R www-data: /var/www/html/wordpress/ /
```

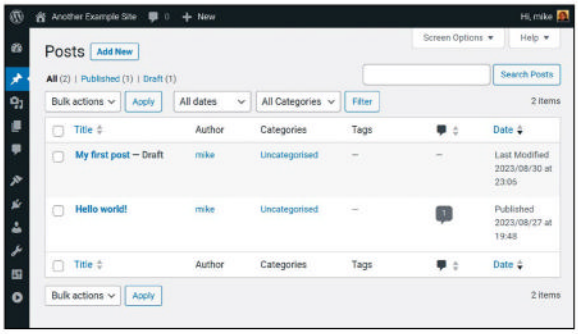
Configure WordPress

There are two ways of configuring *WordPress:* from the command line and using the web interface. You can configure it from the command line by copying **wp-config-sample.php** as **wp-config.php** and editing it in a text editor. However, we're going to use the web interface to complete the *WordPress* installation. One advantage of doing it like this is that it also serves as a test of the setup. Furthermore, the installation script has a bit of intelligence built into it and it can possibly throw out some useful error codes if your setup is missing something that *WordPress* needs.

Log into **http://localhost/wordpress/** in a browser. If you don't see the *WordPress* installation and configuration screen, something has gone wrong with your Apache or PHP setup (but not the *MySQL* setup).

On the next page, we have a page of credentials that *WordPress* uses to communicate with the *MySQL* database. If you've followed the examples that we gave you, these are Database Name: **wordpress_db**, Username: **wp_user**, Password: **wp_pass**. For this setup, the default of **localhost** for the Database Host field is correct. The Table Prefix field is only important if you plan to install more than one instance of *WordPress* to your server. Each *WordPress* installation should have a unique table prefix. Click on Proceed to go to the next stage of the process. Again, if there is anything wrong with your setup, *WordPress* tells you at this point. Assuming everything's OK, click on Run The Installation.

You should now see a page where you can set up the title of the site and the administrator username. If you are the sole user of your *WordPress* site, this user may be the only one that you need. If you are collaborating with other users, *WordPress* has quite a fine-grained user permissions system so that regular users can carry out their work without having full administrative control of the entire site.



The Dashboard is the main configuration area of WordPress. You can also add posts and pages from this area.

You also have to come up with a password. We all like snappy passwords that are easy to remember and type, but *WordPress* sites do get targeted by hackers and you could well get your site hacked if the password is too simple. At the very least, use something that mixes letters and numbers, and is reasonably long.

Log in to WordPress

Hopefully, you see a success screen at this point and you can log into your *WordPress* site for the first time. Once logged in, you are placed at the Dashboard, the main *WordPress* hub of customisation and admin, and you'll be spending a lot of time here. You might like to have a quick look at the site as it stands by clicking on the site title that's next to the home icon that you can find at the top of the window. Once you've examined the default site and the example blog post, you can get back to the Dashboard by clicking on the *WordPress* icon in the top-left corner. In future, if you are logged out of your site, you can always get back to the Dashboard by adding **/wp-admin** to the end of the URL.

Create a post

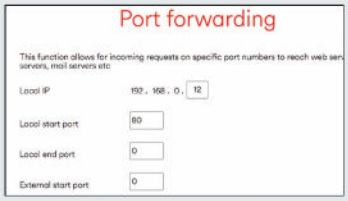
The post and the page are the two basic content types within *WordPress*. Broadly speaking, posts are for news items and pages are for information that you want to have permanently available to visitors; a page about the site itself would be a common example. A post appears on the main blog area of the site. As you add more posts, the older posts move off the bottom page, and are no longer visible on the front page of the site. When you create a page, on the other hand, it is automatically added as a link that is always accessible.

In the spirit of exploring the *WordPress* interface, have a go at creating a post. From the Dashboard, select the Posts tab in the sidebar. From here, click on Add New. You are now confronted with the block-based editor, Gutenberg, a controversial addition that was made to *WordPress* a few years ago. It does away with the normal conventions of text editors as, initially, there are no obvious toolbars to control the text formatting or layout. In Gutenberg, every area of text (such as a paragraph) is a 'block' with given attributes.

To create your first post, simply begin to type in your content, starting with the title of the page. You'll start to get the idea about how things work once you add more paragraphs. To alter text attributes, hover the mouse over the paragraph that contains the text and a toolbar appears. For example, if you want to make a

» ACCESSING FROM THE OUTSIDE

Odds are you want to access your *WordPress* site from outside of your local network. The quickest way of finding your external IP address is to search Google for "show my IP address". If you typed this IP into the web browser, it wouldn't work at first as your broadband router has no way of knowing which machine on your network hosts the *WordPress* server.



A typical router port forwarding configuration page. The exact procedure varies between manufacturers.

Enter the setup admin page of the router. It's normally located at **http://192.168.x.1** (the x is usually a 0 or 1). Every router has its own method for handling admin, so check the instructions for your router. Ultimately, you need to make sure that traffic sent to port 80 is sent to the internal IP address of your server. Type `ip a` to discover this address; it normally takes the form of **192.168.x.y**, with x being a 0 or 1 (same as the router) and y being a number between 0 and 255.

Once you have port forwarding set up, you should be able to browse to your *WordPress* setup from the internet by using a URL such as **http://<your IP address>/WordPress/**.

Adding tags and categories to posts is easy to do and well worth it.

word bold, drag the mouse over that word, hover the pointer over that area, and select the bold icon from the toolbar. Move the pointer away and you'll find yourself with a clean, toolbar-free editing screen again.

If you want to add new content (such as an image, a heading or a list) press Enter to create a blank paragraph. If you now hover the pointer over this empty space, you'll notice a + icon has appeared. Click on this to add a block type to your post.

Get tagging

Turning our attention to the right-hand sidebar for a moment, select the Post tab to attach categories and tags to the post. These facilities are definitely worth using to improve the navigability of your site. *WordPress* tags can contain spaces, so press Enter or add a comma to separate them. Categories are used to indicate the overall content of a post, and tags are used to add detail about topics that are mentioned within the post. Once you have created tags and categories, they can be quickly added to future posts.

The procedure is much the same if you want to create a page rather than a post: start off in the Dashboard, click on Page, and then click on Add New. By default, a link to a new page is added at the top of the main *WordPress* page. However, where links to pages are shown depends on the theme and your own modifications to the layout.

Backing up the database

When disaster strikes, you can often recover a non-functioning site without doing a complete reinstallation, but the method that we're detailing here can be used to add your existing content to a freshly installed *WordPress* instance as well. The *WordPress* site content is stored in the database, and this is what we need to back up. Images and other media are stored in **/wp-content/uploads/folder/** and have to be backed up manually. There are a few things that aren't stored in the database, such as themes, so if the entire site has been lost, these must be reinstalled manually.

We can back up the content from the command line with the following command:

```
$ sudo mysqldump --add-drop-table --no-tablespaces
-h localhost -u wp_user -p wordpress_db >
AugustWPBackup1
```

These are the credentials that we have used in this tutorial. Adjust it for your choices. Note that the **-p** parameter is the database name, not the password. You are prompted for the database password (**wp_pass** in this example) when the command has been accepted. This writes the entire database to a file called **AugustWPBackup1**.

If you open this file in a text editor, you can see that the content is readable. In a dire emergency, you can use this fact to rescue content that you can't get at any other way. Be aware that the content doesn't use line breaks, so might have gone off the right-hand side of the screen.

Restore the database

To restore the database, we first log into the *MySQL* command-line interface as the database administrator as before: `sudo mysql -u root`. Delete the existing database, recreate it as an empty database and exit the *MySQL* command line interface:

```
DROP DATABASE wordpress_db;
CREATE DATABASE wordpress_db;
exit
```

You can then add the backed-up content back into the database from the command line:

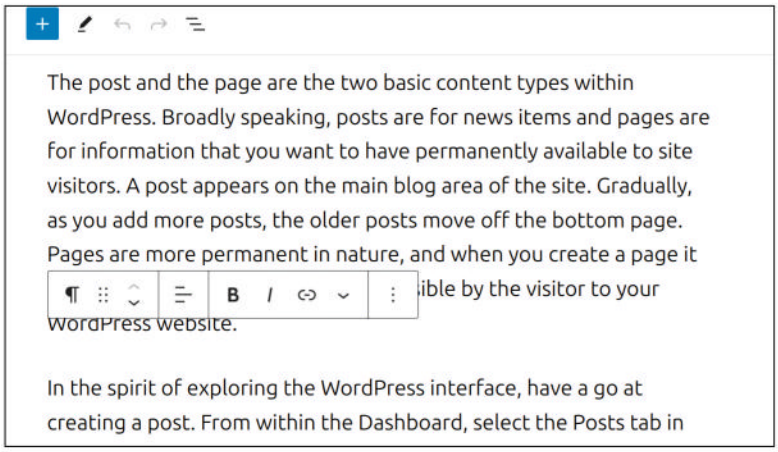
```
$ sudo mysql -u root -p wordpress_db <
AugustWPBackup1
```

This should restore all of your content. If you have a completely non-functioning site, you may have to carry out a complete reinstallation. If starting from scratch again, reinstall the required system components such as PHP and *MySQL*. You also have to recreate the user and grant that user the necessary privileges again. Use the same set of database credentials as during the first installation. Allow the *WordPress* installation to complete as before. Once you have a working site, restore the content into the database.

Going further

Hopefully, we've got you started on your *WordPress* journey. You now know how a *WordPress* website works and how to install it. You also know how to access the dashboard to customise the site in a basic way, and you are able to start adding content in the form of posts and pages.

Next month, we're going to get further into the customisation side of things by exploring layout modifications, themes and templates. LXF



Gutenberg, the WordPress page/post editor. It was controversial on release, but it's a quick way of entering content.

JELLYFIN

Credit: https://jellyfin.org

How to enjoy free streaming TV

Perfect your media server install by adding your own bespoke collection of streaming TV channels. **Neil Mohr** is your channel-hopping guide.

OUR EXPERT

Neil Mohr hasn't got time to watch TV, though he catches the odd *Bluey*, which is a delightful children's TV show and not what you're thinking.

Among the films, TV, books and music, our *Jellyfin* install from the main feature *(page 34)* is looking pretty sweet, but then the boss goes and asks, "What about free TV?" And we wriggle our bums nervously and say "Probably." *Jellyfin,* being a fork of *Emby,* being a fork of *Plex,* does indeed support TV tuners – including scheduling and recording of live TV programmes. While we're not going to specifically cover TV tuner use, we are going to tap into the two systems it adds to *Jellyfin,* which enable us to stream live TV for free from all over the globe.

That's the good news. The bad news is that as what we're doing isn't really a standard thing, it's a bit of fiddle to get up and running. But let's get straight into setting it up. In the *Jellyfin* web client, select the hamburger menu > Dashboard > Live TV. There are two parts to the Live TV system: the tuner device and the TV Guide, aka the electronic programme guide (EPG).

The least important is the EPG. It's just there to add information, not the actual streams or indeed where or how to watch any of it. The issue is where to get the TV Guide XML data from. We found **www.bevy.be/epg-guide** provides XML EPGs for stations around the globe. We're going to add UK stations, so we're using **www.bevy.be/bevyfiles/unitedkingdom.xml**. Copy this URL, then in *Jellyfin* click the TV Guide + icon > Select XML, paste it into the URL section, then click Save.

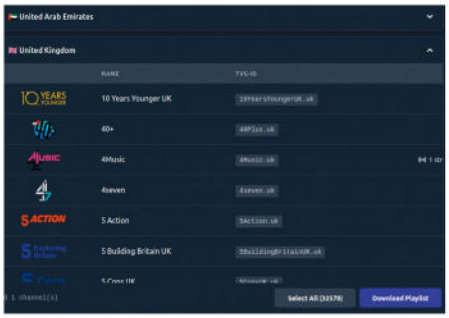
Note: if you click on the Bevy Channel list for that country, the name of the channel you want has to match the name we're about to add to our config file.

Tuning in

How do we add a non-existent TV tuner? The system is set up to handle IPTV streams where the channels are defined in an M3U file. Yes, these are technically the *Winamp* audio playlist files from a bygone era. They were co-opted by the streaming media world, and are just text files with metatags to define each stream.

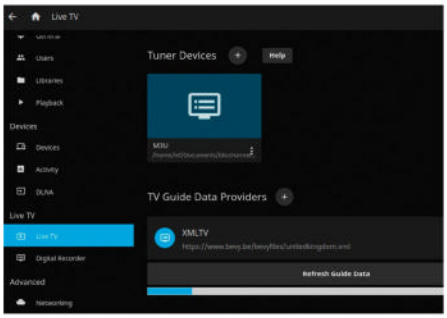
When we were originally playing around, we were writing these by hand, but luckily we've found a way to

ADD IPTV TO JELLYFIN



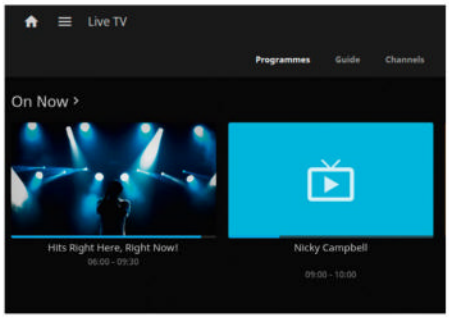
1 Create a playlist

Covered in more detail in the main article, head to **https://iptv-org.github.io**. Click Playlist and select the channels you want listed in this collection. Download the file and edit the list to ensure the channel names match those within your EPG file from **www.bevy.be**, if that's what you're using. Move it to **Documents**.



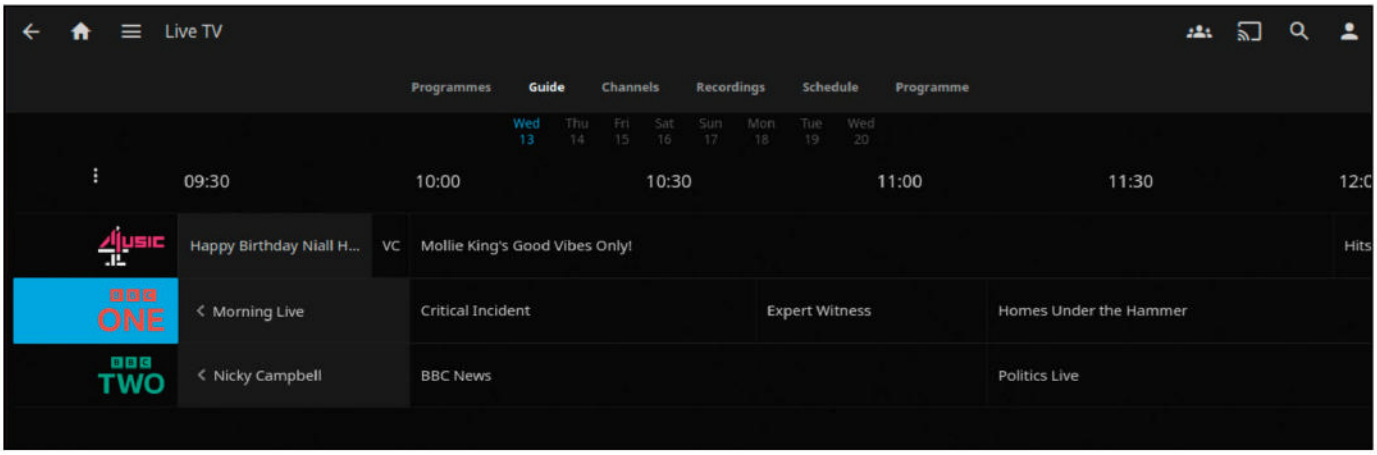
2 Set Jellyfin

In the *Jellyfin* web client, click the hamburger menu > Dashboard > Live TV. Under TV Guide Data Provider use the + to add the EPG XML address. Click the Tuner Devices + button. From Tuner Type select M3U Tuner. Click the spyglass and locate the M3U file; for example **/ > home > <user name> > Documents**. Click OK, then Save.



3 Watch TV!

You should spot the Refresh Guide Data being automatically refreshed – that's the blue bar. Click the Home button and a new Live TV section should appear under My Media alongside a Live TV selection – with various Programme/Guide buttons – and On Now. Click one to enjoy your new streaming TV!



A fleshed-out Jellyfin TV guide, bristling with endless streaming TV options.

Jellyfin can handle any sort of stream you throw at it, even chat radio.

generate playlists of any channels you like. So, you can have multiple M3U files, each assigned as a separate tuner and each file can have multiple channels within it.

The easiest way to create an M3U 'playlist' file is to head to **https://iptv-org.github.io**. Behold! A website that lists every single streaming channel in the world and the important information about each address.

It's organised by country, so we'll head to the UK and open that section. In the far-right column, keep an eye out for Stream. If a channel has an accessible digital steam, that shows up here. Take BBC One England, click Stream, open up the two sections within and you'll see the raw addresses. One ends in .mpd and the other address ends with .m3u8. MPD is the higher-quality MPEG-DASH format while M3U8 is typically the older format .ts Transport Stream format.

Thankfully, we don't have to worry about these too much. Click Create Playlist – this creates selection buttons beside every channel (we had to engage dark mode to see these clearly, but that's probably just our age). Select all the channels you want – we'd keep it simple to begin with – and hit Download Playlist.

Open this file and have a nose around. Each channel starts with a **#EXTINF:-1** tag. The **tvg-logo** helps add the channel logo to your EPG. The **group-title** adds a subcategory type, such as news or music, but the important one from the EPG point of view is the name after the last comma. This needs to match the name in the EPG file, as mentioned earlier – it's a good idea to tweak these at this stage, so they're correct. After this on its own line is the streaming URL address.

Name the file something more obvious and move it to somewhere suitable, such as **Documents**. Switch to the *Jellyfin* web client and click the hamburger menu > Dashboard > Live TV. Click the + next to the TV Tuner. Under Tuner Type, select M3U Tuner and click the spyglass icon at the end of the File or URL box. Locate the M3U playlist file you saved. If you put it in **Documents** this would be **/ > home > <username> > Documents > <playlist>.m3u**. Click OK, then Save, and it never hurts to click Refresh Guide Data.

If you're running *Jellyfin* from the terminal, you should see a ream of log updates about EPG updates and the channels you've added being processed. Click the Home icon and hopefully there is now a Live TV Media section, a Live TV list and perhaps an On Now section. It's worth clicking Channels as this is the most basic list of channels you've added. Click the logo of one and it should start to play. If you've paired up the EPG correctly, when you click Guide you should see scheduled programmes for each channel. Click on a programme and you can set up recordings, too. LXF

QUICK TIP

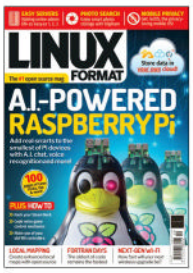
If you fancy using the TV tuner abilities, you need to grab an HD Homerun TV tuner card – though at £150 they're not the cheapest bits of kit now.

» PREBUILT OPTIONS

We've covered *Jellyfin* in detail as that's the media centre option we've covered in the main feature, so it made sense. But its lack of pre-built IPTV channels is a little annoying – though, hopefully, the minimal tinkering we've outlined here isn't too onerous. However, there are plenty of ways to skin a cat. The elephant in the room is *Kodi*. If you're after streaming TV options, *Kodi* remains the premier option for TV-connected Android or any other OS, frankly. Check the Download section at **http://kodi.tv** – most Linux users can get it via Flathub. Once running, select Add-ons > Download > Video Add-ons, and choose something to stream. *Kodi* is a little heavyweight but the sheer numbers using it means it's better supported than most other options. Another choice is *Plex* (**http://plex.tv**). You can sign up and install the media centre for free, part of which is a selection of curated streaming TV channels, though it is pushing its premium subscription service to people.

BACK ISSUES » MISSED ONE?

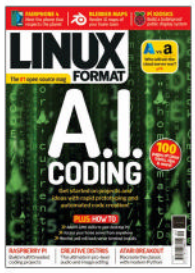
ISSUE 307
October 2023



Product code:
LXFDB0307

In the magazine
Power up your Pi with awesome AI projects, and take control of your phone with the privacy-focused /e/OS. Plus, find out which open source personal cloud server is best for you, read all about the forthcoming Wi-Fi 7, and discover how to transform images into GIFs via the terminal, manage a VPS, boost your Steam Deck, and much more. And there's our usual news, reviews and sysadmin tips.

ISSUE 306
September 2023



Product code:
LXFDB0306

In the magazine
Discover how AI can (and can't) help with your coding, plus add Pi-style GPIO to your PC, access your home server from anywhere, and write your own text adventure and *Breakout*-style games. Also, *Adminsteria* returns with news and reviews for sysadmin types, we take an in-depth look at how Azure stacks up against AWS, there's a *Roundup* of media-creation distros, and plenty more...

ISSUE 305
Summer 2023

Product code:
LXFDB0305

In the magazine
Don't let Big Tech be in control of your documents – find out how to run your own office apps in the cloud. Plus, learn how Linux certification can kick-start your career, step back in time with our look at the evolution of coding, and read our recommendations for the best open source CAD packages, not forgetting our usual packed tutorial, review and Raspberry Pi sections.

ISSUE 304
August 2023

Product code:
LXFDB0304

In the magazine
Run your own AI and take over your world with local open source machine learning, discover the diverse delights of BlendOS, and learn all about AMD, Intel and Apple's different takes on CPU design. Plus, we have an in-depth interview with Collabora's Michael Meeks, a *Roundup* of official Ubuntu spins, and tutorials on everything from restoring old photos to coding a text adventure.

ISSUE 303
July 2023

Product code:
LXFDB0303

In the magazine
Trim down your Ubuntu installation for a stripped-back, super-slick and ultra-fast Linux experience. Plus, discover which beginner programming language is the best, read all about OpenSUSE's ALP distro, and find out why *Podman* might be better than *Docker*. And don't forget to try our tutorials on terminal-based to-do lists, searching file contents, making mazes in C, and loads more.

ISSUE 302
June 2023

Product code:
LXFDB0302

In the magazine
Master hacking with Kali Linux, discover the artful ways of *Stable Diffusion*, try a taste of the anything-but-plain Vanilla OS and find out what's new in the online office world. Plus, pick up a whole bunch of new skills thanks to tutorials on speeding up text-based tasks, rescuing data from floppies, repairing images in *GIMP* and much more, and learn which is the hottest firewall distro in our *Roundup*.

To order, visit www.magazinesdirect.com

Select **Single Issues** from the tab menu, then select **Linux Format**.

Or call the back issues hotline on **0330 333 1113**
or **+44 (0)330 333 1113** for overseas orders.

Quote the product code shown above and have your credit or debit card details ready.

COBOL

COBOL: still going after all these years

It might have appeared back in 1959 and look odd compared to today's languages, but COBOL is still fuelling business, **Mike Bedford** reports.

OUR EXPERT

Mike Bedford has impressive knowledge of programming languages, but this month he's turned his hand to natural languages, too.

Following our recent exposé of several classic programming languages – most of them largely forgotten or quirky – we delved into Fortran last month. Despite its age, the name is still familiar among coders and, although it was primitive in its initial guise, it would be wrong to describe it as peculiar. In fact, it was influential in the development of some of today's most popular languages. Not only that, but it still plays an important role today. However, Fortran isn't alone in that respect.

Here we're delving into another positively ancient language that continues to play a key role in so many areas. And to illustrate that the language genuinely is from a long-gone era, we only have to mention the names of the computer manufacturers who were involved in its joint development. Of course, IBM played a part, but so did Burroughs, Minneapolis-Honeywell, RCA, Sperry-Rand and Sylvania, all now barely remembered as computer companies. That language is COBOL – COmmon Business Oriented Language – and while it's only two years younger than Fortran, first seeing the light of day in 1959, in most other respects it couldn't be more different.

One of the two computers on which COBOL first ran, the Sperry-Rand UNIVAC I is the stuff of history books but the language lives on.

Coboled together

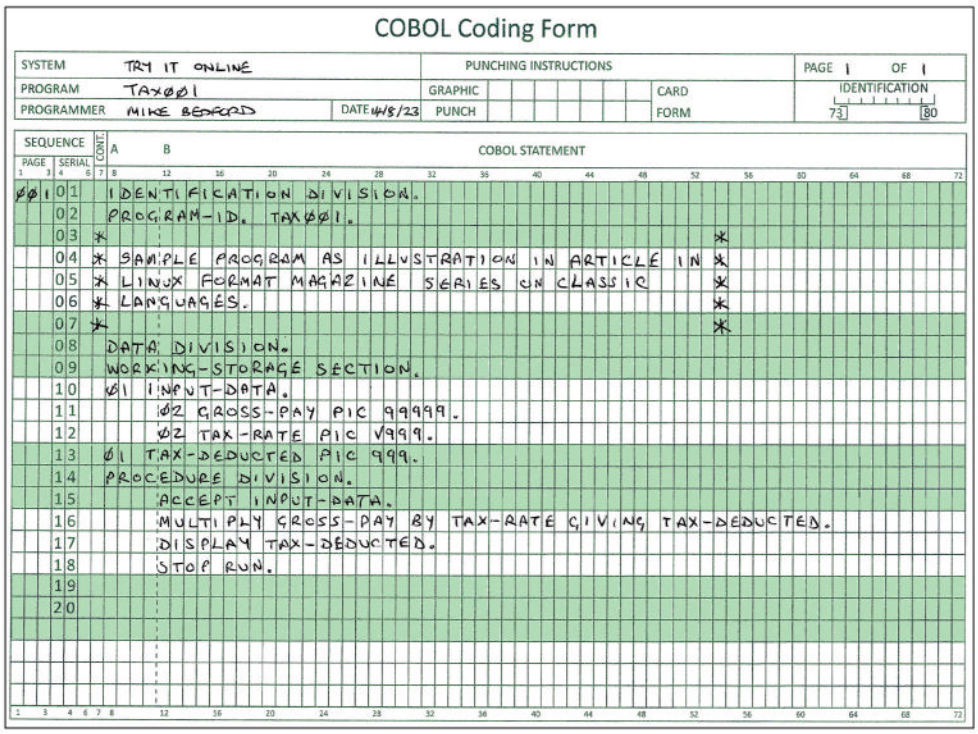
A survey of the people who have brought us the hardware and software developments that shaped modern life tends to reveal job descriptions such as engineer, physicist, academic, mathematician, computer scientist or entrepreneur. Rarely do we come across the description rear admiral, United States Navy, but that's all about to change as we introduce Grace Hopper. OK, we do have to admit that she was also a computer scientist and mathematician, but Grace Hopper is considered the key player in the development of COBOL, a joint project involving the US Navy, the US Air Force, the Bureau of Standards and a cross-section of computer manufacturers. The aim of the project was to develop a language for business applications that would be portable across platforms. This was in stark contrast to the scientific, technical, engineering and mathematical applications that Fortran had catered for. Hopper's vision, which was indeed embodied in COBOL despite opposition in some quarters, was for programs to be written in the English language, rather than in a terse mathematical notation. So, for example, a statement that would appear as `TD = GP * TR` in Fortran would become `MULTIPLY GROSS-PAY BY TAX-RATE GIVING TAX-DEDUCTED.` in COBOL. Yes, we could have used more meaningful variable names in the Fortran code, and that would have made the two bits of code look more similar, but having said that, it was common practice to use terse variable names in Fortran, but longer names were encouraged in COBOL.

Grace Hopper's assertion was that it's easier for most people to write an English statement than to use symbols. However, after a few years, we tended to hear, instead, that the benefit was not so much to the programmer as to non-programmers who might need to read and understand the code. It was suggested, for example, that there would be occasions on which accountants, auditors and financial directors would need to understand the programs that drove their organisations' systems. It's been suggested, though,

Sometimes referred to as the grandmother of COBOL, Rear Admiral Grace Hopper of the US Navy was part of the team that developed the language in the late '50s.

CREDIT: Public Domain, www.census.gov/history/ Public Domain, www.history.navy.mil

that the reasoning is flawed. Sure, the COBOL statement we showed is slightly easier to understand than the equivalent in Fortran, but it surely doesn't take long for a non-mathematical non-programmer to understand individual statements. Much more difficult, and something that COBOL didn't address, is understanding the program structure and operation as a whole. Indeed, this is something that even programmers struggle with on attempting to maintain or enhance a program written by a colleague. And it gets worse. English-like statements result in code that can be described as verbose and this, it's been suggested, tends to hinder interpretation. And since it takes longer to write long statements than short ones, we rather wonder if the COBOL approach actually reduces programmer productivity.


COBOL Coding Form

SYSTEM: TRY IT ONLINE
PROGRAM: TAXØØ1
PROGRAMMER: MIKE BEDFORD
DATE: 4/8/23
PUNCHING INSTRUCTIONS
GRAPHIC PUNCH
CARD FORM
PAGE 1 OF 1
IDENTIFICATION 73 80

SEQUENCE PAGE SERIAL CONT. A B COBOL STATEMENT

```
ØØ1 01   IDENTIFICATION DIVISION.
    02   PROGRAM-ID. TAXØØ1.
    03   *
    04   * SAMPLE PROGRAM AS ILLUSTRATION IN ARTICLE IN *
    05   * LINUX FORMAT MAGAZINE SERIES ON CLASSIC      *
    06   * LANGUAGES.                                   *
    07   *                                             *
    08   DATA DIVISION.
    09   WORKING-STORAGE SECTION.
    10   Ø1 INPUT-DATA.
    11      Ø2 GROSS-PAY PIC 99999.
    12      Ø2 TAX-RATE PIC V999.
    13   Ø1 TAX-DEDUCTED PIC 999.
    14   PROCEDURE DIVISION.
    15      ACCEPT INPUT-DATA.
    16      MULTIPLY GROSS-PAY BY TAX-RATE GIVING TAX-DEDUCTED.
    17      DISPLAY TAX-DEDUCTED.
    18      STOP RUN.
    19
    20
```

Originally, COBOL code had to adhere to a fixed format, with different types of statement starting in different columns. Coding forms helped programmers use the correct columns.

So, the benefits or otherwise of interchanging symbols for English words might be debatable and the fact that few, if any, more recent languages have followed that lead might be telling. However, other features of COBOL, which it didn't share with Fortran or most other early languages, definitely did favour financial and business applications. Top of the list is support for hierarchical data structures, which shares some similarity with C's structures, but it would be another 13 years before that language made an appearance. These were commonly used in reading fixed format data, for example from punched cards or fixed format disk files, and would define how the data was presented in each record. So, for example, a customer might be defined by a name, address, date of most recent purchase, and so on, and some of these variables would be subdivided. Any dates, for example, would be divided into day, month and year. And the subdivisions might contain further subdivisions. All the lowest levels would be defined by the type of data (such as numeric or text) and its length. Let's consider an example to see how being able to refer to any level in a data structure improves efficiency. A classic COBOL example is data validation in which data is read from a stack of punched cards – the data on those cards found to be valid is written to file for subsequent use, and error messages identifying invalid cards are printed out. So, for example, validating a date involves checking that the month has a value between 1 and 12, which involves accessing a bottom-level entry in the data structure. However, writing a valid record to a data file could be achieved using a single instruction that referred to a top-level entry.

Hands-on

We assume most of you won't decide to do any serious coding in COBOL, but you might like to try it briefly to better appreciate its unique nature, so we're only going to provide a trivially simple example. You could try this, and perhaps enhance the code, using the online facility Try it Online (**https://tio.run**), but be sure to take a look at the notes that follow the code first. Given its niche nature, at least for non-professionals, you're probably not going to want to install a compiler locally, but if you

» COBOL AND THE Y2K BUG

Back in 1999, the more sensationalist news outlets reported that civilisation would end with the dawn of the new millennium. Some carried reports of individuals who planned to escape Armageddon by stocking up on provisions and guns, and abandoning the cities for the safety of the wilderness.

This related to the Y2K bug that would, it was suggested, cause computer systems worldwide to fail at the stroke of midnight on Millennium Eve. Failures were reported, but widespread disruption was avoided. The problem was that old COBOL code stored dates with a two-digit year instead of four, so the year 2000 would be interpreted as 1900 – such was the imperative of saving two bytes every time a date was stored in those days of sky-high storage costs. Many companies pre-empted the issue by recruiting armies of COBOL programmers, many brought out of retirement, to track down and correct code that wasn't millennium-compliant. History suggests they were successful in their quest but, in hindsight, not everyone saw it that way. For some, that sigh of relief when the world didn't end was soon replaced by the view that the whole thing had been a hoax. Try telling that to Wayne Horscroft, one of the COBOL programmers who beavered away on the problem in the late '90s, heading the year 2000 project for Utah Power & Light. Speaking to the *Los Angeles Times*, Horscroft admitted that the stress and hectic pace of the job caused him to take early retirement.

Recent versions of COBOL, while still having English-like syntax, allow code that looks much more similar to modern languages.

```
procedure division.
    display "Enter a calendar year (1601 thru 9999): "
        with no advancing
    accept yr
    if yr >= 1601 and <= 9999
        continue
    else
        display "Invalid year"
        stop run
    end-if
    perform 12 times
        move 1 to da
        add 1 to mo
        if mo > 12              *> to avoid y10k in 9999
            move 12 to mo
            move 31 to da
        end-if
        compute int-date = function
            integer-of-date (rd-date)
        if mo =12 and da = 31   *> to avoid y10k in 9999
            continue
        else
            subtract 1 from int-date
        end-if
        compute rd-date = function
            date-of-integer (int-date)
        compute dow = function mod
            ((int-date - 1) 7) + 1
```

do, we suggest *GnuCOBOL*, which just happens to be the complier hosted at Try it Online.

```
    IDENTIFICATION DIVISION.
    PROGRAM-ID. TAX001.
    DATA DIVISION.
    WORKING-STORAGE SECTION.
    01 INPUT-DATA.
      02 GROSS-PAY PIC 999999.
      02 TAX-RATE PIC V999.
    01 TAX-DEDUCTED PIC 999999.
    PROCEDURE DIVISION.
      ACCEPT INPUT-DATA.
      MULTIPLY GROSS-PAY BY TAX-RATE GIVING
TAX-DEDUCTED.
      DISPLAY TAX-DEDUCTED.
      STOP RUN.
```

Because of its unusual nature, especially for 21st-century programmers, the following notes will help you to better understand the code.

1. The original COBOL placed strict restrictions on which columns to use. This has been relaxed more recently, but don't start any line before column 8.

2. A COBOL program is split into divisions, initially the **IDENTIFICATION**, **ENVIRONMENT**, **DATA** and **PROCEDURE DIVISIONs**, although most of today's compliers require only the first and last of these. The **IDENTIFICATION DIVISION** contains a mandatory **PROGRAM-ID**, but additional types of information can be provided. The **ENVIRONMENT DIVISION**, which we've omitted, contains information that relates to specific hardware so, in theory, only this division might need changing on porting the code. Included here, for example, are details of I/O devices. The **DATA DIVISION**, which, despite being optional, is required in virtually all applications, contains the equivalent of variable declarations, albeit tailored to support hierarchical data structures. And the **PROCEDURE DIVISION** contains executable statements.

3. If our code had contained disk-based I/O, the **DATA DIVISION** would have included an **INPUT-OUTPUT SECTION**. However, since we're only using **ACCEPT** and **DISPLAY** statements for I/O *(see point 5)*, this isn't needed, so we only have a **WORKING-STORAGE SECTION**, which defines 'ordinary' variables. We have just two top-level (01) variables, namely **INPUT-DATA** and **TAX-DEDUCTED**. Against **TAX-DEDUCTED** there's a **PIC 999999**, which defines the variable as six-digit numeric. There's no such definition against **INPUT-DATA** because it's divided into two lower-level (02) variables. The first of these is defined as a six-digit number, and the second is a numeric value comprising a three-digit number with implied decimal point at the start.

4. Statements are terminated by full stops to further emphasise the English language nature of COBOL.

5. To simplify things, we're using **ACCEPT** and **DISPLAY** for I/O. In the early days, these would have read or written from the operator's console, but they wouldn't commonly have been used. However, unlike other forms of I/O, such as **READ** or **WRITE**, we don't need to provide information in an **ENVIRONMENT DIVISION** or an **INPUT-OUTPUT SECTION** of the **DATA DIVISION**. **ACCEPT** can only accept a value for a single variable, so it's the same as reading from a punched card in that respect. That variable is **INPUT-DATA**, which is divided into **GROSS-PAY** and **TAX RATE**, the latter of which isn't a percentage but a value in the range 0 to 0.999. These two values are entered immediately after each other, and should be typed into the **INPUT** area before executing the code if you're

» COBOL TRANSLATIONS

As one of COBOL's aims was that code should be English-like so it could be read by non-programmers, it's interesting to consider whether there were variants based on other natural languages. After all, a French accountant wouldn't find an ordinary COBOL program to be as easily readable as a British or American accountant. Well, it transpires that there were, indeed, translations of COBOL.

We'd like to be able to report that we found something as bizarre-looking as Greek or Arabic COBOL code, but we're not sure that such a thing ever existed, and we can't even conceive of a Chinese version. We do understand, however, that COBOL was translated into Russian, and that, of course, involved the Cyrillic alphabet. Like most things relating to the former Soviet Union, though, details are almost impossible to find. However, back in 1965, the European Computer Manufacturers Association published a report defining the translation of COBOL into French, German and Italian, and these were later published as national standards. So, if you thought that the original COBOL was odd, you might be interested in these equivalents to our previously quoted instruction `MULTIPLY GROSS-PAY BY TAX-RATE GIVING TAX-DEDUCTED.`. In French COBOL, this would be `MULTIPLIER SALAIRE-BRUT PAR TAUX-DIMPOSITION RESULTANT TAXE-DEDUIT.` (yes, we also took the liberty of translating the variable names), while in German it's `MULTIPLIZIERE BRUTTOLOHN MIT STEUEERSATZ ERGIBT STEUERABZUG.`, and the Italian equivalent is `MOLTIPLICA RETRIBUZIONE-LORDA PER ALIQUOTA-FISCAL DANDO IMPOSTA-DETRATTA.`.

CREDIT: Public Rosetta Code

using Try it Online. So, for a gross pay of 40,000 and a tax rate of 0.250, you'd enter `040000250`.

Evolution

As a language that's still in use after more than 60 years, it's not surprising that, like Fortran, COBOL has gone through many iterations, the most recent being COBOL 2014. New versions came close on the heels of the original; as early as 1961, a new version appeared providing features that, arguably, should have been in the original. It seems that the features in the initial version of COBOL were compromised because of the ambitious development timescale. Despite this, the new instructions in that first update were innovative. With the exception of its hierarchical data structure, the features in the original COBOL were much the same as those in other contemporary languages, albeit with its unusual English-like syntax. However, COBOL 61 provided instructions that would probably not even have been considered for languages intended for scientific or technical applications. First of these business-specific instructions was the **SORT** statement. Its purpose is to sort an unsorted input file, writing the result to an output file. The statement defines one or more keys – variables at a lower level than the top-most file record variable – each of which can be sorted in either an ascending or descending order. Without the **SORT** statement, it would be necessary to write a not-insignificant amount of code, or else use an external library function with a possible negative impact on portability. Next up is the **MERGE** statement, which reads data from two or more previously sorted files, writing a merged file according to a sort specification. Then there were various instructions to simplify report writing.

Like Fortran, COBOL has also benefitted from upgrades that introduced programming conventions and paradigms that appeared after it was first devised. The upper-case-only requirement – imposed because of the limitations of punched cards and line printers – was relaxed in favour of upper and lower-case letters. The block-structured approach was introduced. And finally in our whistle-stop tour, starting with COBOL 2002, COBOL could boast object orientation.

COBOL today

Given its age, we might expect the use of COBOL to have diminished spectacularly since its heyday, but the facts tell a different story. As recently as 2022, key COBOL development tool supplier Micro Focus, now part of OpenText, commissioned a survey that revealed some remarkable facts. More than 800 billion lines of COBOL support today's core business systems, and 52% of respondents predicted that COBOL will be here for the next decade or beyond, while 83% believe it will be there well after they retire, and 54% expect COBOL use to increase in the next 12 months. No fewer than 92% said their COBOL applications were strategic.

These statistics aren't indicative of code that's static and failing to keep up with modern requirements. That huge stack of vintage code – much of it dating back to pre-internet days – isn't just being maintained; indeed, that survey also showed that 72% of respondents preferred modernisation as an overall business strategy, and two out of three said they intended to modernise their COBOL applications rather than replace them because it's strategically the best option for the risk-averse enterprise. But not only has COBOL remained critical to business, it's also providing consumers, with so much of what we rely upon, as OpenText's Ed Airey explained to *TechBeacon*. "Most of the things that we just frankly take for granted every day, like being able to take money out of the bank using an ATM or book an airline ticket for vacation or get an insurance quote, depend upon COBOL running quietly in the background," he said. This, of course, relies on modernisation, to embrace technologies that were unheard of when the code was first written, as Airey explained to *ITPro Today*. "The overall COBOL codebase continues to evolve and change as new IT enterprise capabilities such as web and mobile access, cloud deployment, API integration and containerisation are leveraged in support of new customer requirements. It's clear from the survey results that COBOL is not standing still nor in maintenance mode, but rather embracing new change in support of a new digital era." We'll give the last word to OpenText's Scot Neilson, also talking to *ITPro Today* about COBOL's place in the modern world. "It's still here because all the stuff people created decades ago can still run today. You don't have to throw it away and start again. You can build on top of it."

Might the world still be singing COBOL's praises in another 60 years? Perhaps not, if a development recently announced by IBM lives up to its potential. Called WatsonX Code Assistant for Z, IBM says this generative AI-assisted product will help enable faster translation of COBOL to Java on its Z mainframes, and enhance developer productivity on the platform. So, the writing might be on the wall for COBOL, but would you really be prepared to bet on it? Or might the pundits end up quoting Mark Twain, who famously said, on hearing reports that he'd breathed his last, "The reports of my death are greatly exaggerated"? LXF

Even today, it's common for COBOL code to run on mainframes, like this IBM Z.

DOMESDAY86

Credit: www.domesday86.com

Reviving the BBC Domesday Project

Matt Holder discovers how the BBC's Domesday Project was nearly lost to recent history and how a group of volunteers are bringing it back to life.

OUR EXPERT

Matt Holder has worked in IT support for over a decade and always tries to utilise Linux alongside the other installed systems.

The *Domesday Book* is an incredible piece of British history, compiled between 1085 and 1086 as a record of the Great Survey ordered by King William I. It covered land usage and money owed to the king. Much of England and parts of Wales were surveyed. The book survives today and is stored at the National Archives at Kew Gardens.

What does this have to do with us computer lovers? Well, during the '80s, the BBC launched a project to mark the 900th anniversary of the book, because part of the BBC's remit is not only to broadcast TV programmes, but also to educate viewers. The project aimed to be a survey of sorts about the whole of the United Kingdom, with much of the information produced by school children.

The BBC, Acorn, Phillips and Logica worked together on the project because some serious (for the time) hardware and data storage was needed.

Dredging his memory, this writer still remembers a day in year five of primary school when a trolley was wheeled into the classroom. On this trolley was a BBC

The complete Domesday system setup.

Master AIV, a monitor, which was pretty much the shape of a perfect cube, a LaserDisc player and a trackerball. The trackerball was used to navigate around the software's interface, which seemed positively space-age at the time. From the front of the classroom, Mr L announced that the day's lesson would

CREDIT: Domesday86 Project

» THE DOMESDAY86 PROJECT

First, as way of introduction, please meet Ian *(left)* and Simon *(right)* from the Domesday86 project. Your author has known Ian for around 20 years. We're also chatting with Chad, the core developer behind *ld-decode*.

LXF: When did you first become interested in the BBC's Domesday Project?

IAN: The local education authority loaned one to the secondary school that I attended. I remember seeing it but the teacher in charge said that I could not use it and that "You will never be able to afford one." That's where the challenge started. The first time I used one was at the Science Museum in London, and it was fascinating just flipping through the maps of the UK.

In about 2005, I managed to find a set of Domesday discs for sale and decided I would try to find all of the parts, which took almost a decade to get my first working setup.

SIMON: About seven years ago, I wanted to have a play with the Domesday system; I'd read about it, but never used one. I asked on a forum for a copy and was met with mostly radio-silence. Ian was the first to come back to me and explain that the LaserDiscs can't be copied like CDs. At that point, I'd never even used a LaserDisc, so I had no idea.

LXF: What made you decide to work on the archival of data from the project?

IAN: When I started university in later life, one of the first lectures was about keeping backups of your work and data. The BBC Domesday Project was an example of how not to store data, with a headline of "Domesday lost after 20 years". There was shock in the auditorium when I put my hand up and announced that I owned the discs and the necessary hardware to play them.

SIMON: Curiosity. It's easy to make high-quality copies of anything if you just throw money at it but we needed a

encompass a demonstration of the *Domesday* software, presumably loaned by the local council. A giant silver-coloured disc was then taken from its protective case and loaded into the player.

Laser trailblazer

LaserDiscs were 12 inches in diameter and stored analogue video and digital or analogue sound. Depending on the method used, disks could hold between 30 and 36 minutes, or between 60 and 64 minutes of video per side, in a quality higher than that offered by VHS or Betamax. The discs were read by a laser (much like a CD or DVD) and the information was encoded using frequency modulation – think of FM radio as a comparison. Using various techniques, features such as slow motion and freeze frame were made available to the user. LaserDisc was more popular in Japan than Europe and the US. Lessons were learned and CD, DVD and Blu-ray all built upon what was used for LaserDisc.

The BBC Micro and Master computers were a collaboration between Acorn and the BBC, and worked alongside a set of TV programmes and resources, set around educating the public about computing. For the time, these devices were incredibly customisable, with expansion slots being available for usage. For the Domesday Project, the BBC Master was fitted with a SCSI interface and a second processor, which was primarily used to control the LaserDisc player. The software was written in BCPL, which was a precursor language to C. The choice seems to have been to enable the software to run on multiple architectures. Sadly the lack of popularity meant that ports were not readily made, but there was a version that ran on the RM Nimbus x86 computer.

Now we've discussed the hardware, let's take a look at the information that was gathered for the survey. The Community Disc is one of the original two and contains information provided by the public in the form of photos and reflections. The area surveyed was split into 4x3km areas, to match the ratio of the monitors the software was designed for with space allocated for three photos and a small number of reflections. This information was navigated from a map of the UK. The second original disc is the National Disc, which contains more varied information, such as data from the 1981 census, professional photos and virtual reality views of spaces. Some of the material was browsed using a virtual art gallery space, where objects on the walls could be selected and doors opened to move between different areas of the gallery.

Other content also released included the Ecodisc, which covers nature and ecology, and uses information set on a nature reserve. Instead of using LaserDiscs, the North Pole expedition contained data that was

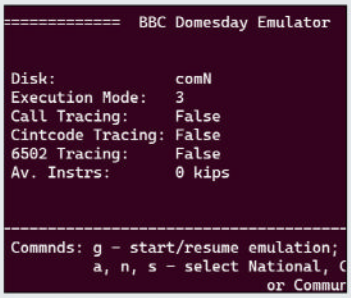
» A PYTHON APPEARS ON DOMESDAY

It is possible to emulate the Domesday hardware by using *pyDomesday*. The software emulates the BBC Master and associated hardware, and uses Pygame to render the images on the screen. Installation is simple – follow the steps below:

```
$ mkdir Domesday
$ cd Domesday
$ git clone https://github.com/stevem2/pyDomesday.git
$ cd pyDomesday
```

Once you've moved to the directory, you can edit **BeebCfg.py** to point to the data files that are required. Once edited, run **Beeb.py** and follow the on-screen instructions.

Unfortunately, due to copyright issues, it is not possible to safely and legally download the files. It is up to you to determine whether you wish to take the risk. The Python script requires an ADL file, which contains a disc image for booting as well as content images/video files. There is a safer way to view the information captured for Domesday – see the main article.



The emulator gives options for selecting disks and changing settings.

QUICK TIP

The online emulator for Domesday can be found at https://domesday.computinghistory.org.uk.

low-cost way. By using a real LaserDisc player to spin the disc and track the playback, I could use a cheap sampling system to grab the optical data direct from the laser.

CHAD: Ian and Simon came to me with the project, and saving these discs sounded fun! It led to pretty much redoing everything I had up until that point and everything came out much nicer.

LXF: This sounds like pretty technical work – is your background working on this sort of technology?

IAN: I have backgrounds in both land surveying and software engineering. Alongside this, my mother worked for a local archives office, so I think that it's one of the few topics that encompasses all three in one form or another.

CHAD: I'm a software engineer who works with real-time embedded OSes, which means signal processing isn't my main background.

SIMON: I have a long history of working with both software engineering and electronics. I started my career as a software engineer, but hardware and electronics have always been a hobby. For my real job, I work as a sales engineer – quite far removed from my hobbies.

LXF: In simple terms, what do you feel is the most interesting part of the Domesday Project?

IAN: It is stunning how so many pieces of new technology were contained within one package: the SCSI-controlled LaserDisc player that contained video mixing, the Domesday discs with multiple data types stored, and the BBC Master AIV – a dual-processor BBC Master for display and control. The software was written in BCPL (a precursor to C) that compiles to Cint code and runs on the equivalent of a virtual machine in the same way as Java or .Net.

SIMON: The content of the Domesday LaserDiscs – when accessing the Community Discs, you can almost hear the children's voices in 1986 talking about whatever they were interested in and, in a lot

QUICK TIP

More information about *ld-decode* can be found at https://github.com/happycube/ld-decode.

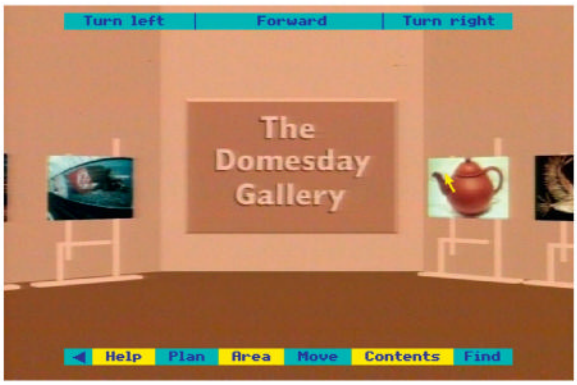
stored on LaserDisc and software that ran from floppy disks. The last two were the Countryside Disc, which dealt with agricultural data, and the Volcano Disc, which covered the science behind volcanoes.

We suspect that asking for a hands-up of anybody who has used a LaserDisc and player would result in a very low number. This completely unscientific statistical analysis probably tells us the main reason why the Domesday Project wasn't as much of a success as it should have been and also why the data nearly ended up being lost to recent history.

Unfortunately, when the discs were donated to their new home, the archivists don't seem to have been particularly proactive with extracting the data and the LaserDiscs were largely left alone. Over the years, the discs degraded, meaning that data extraction was incredibly difficult and a number of retrieval projects sprang up. A new project, Domesday86, was started by a couple of volunteers, who felt that the archival of this data was incredibly important. Huge amounts of information about the Domesday Project and the data retrieval work can be seen at **www.domesday86.com**.

A piece of custom hardware has been developed by the team, known as the Domesday Duplicator. This connects to the RF (radio frequency) output of specific LaserDisc players, samples the data and allows it to be saved to a hard drive or SSD by connecting it to a USB port on a PC. This RF sampling is important because it enables all data on the disk to be accessed – the Domesday discs contain different data items, which sampling as audio visual outputs from the player would not be able to reproduce. The

Navigation of the Community Disc was via a map view.



The National Disc can be navigated by walking through a virtual gallery.

Domesday Duplicator is made up of three boards: a USB 3 development board, an ADC (analogue-to-digital converter) board and an FPGA (field programmable gate array) board, which processes the data from the ADC and then enables it to be sent to the PC, via USB.

Decode the degraded

This hardware is excellent, but how does raw RF information get converted into text, video, audio or image data? Well, a piece of software called *ld-decode* was written by somebody outside of the Domesday86 project. Without this tool, the work that has been done would not be possible. This is open source, released under GPLv3, and is written in Python. Its purpose is to emulate a LaserDisc player and do everything required to convert the data into the necessary file formats. Looking into the documentation, the software does an amazing number of things, such as filtering the data as well as format conversion.

Given the analogue storage of the data and the fact that each set of discs degrades differently, the converted files are all slightly different. Work is ongoing to take converted data from multiple discs, combine it together and create a data set that contains the best information from multiple sources.

The way the trackerball operates also happens to be different from the protocol used by modern mice. The

CREDIT: Domesday86 Project

» THE DOMESDAY86 PROJECT *CONT.*

of cases, you can feel the teachers in the background desperately trying to ensure they make a good impression.

LXF: First of all, thank you, but what made you decide to open the work you are doing to the community, by using open source licences?

IAN: By giving everyone the tools to be able to duplicate their own discs and then decode the contents, there is a higher chance of the data surviving for the next generation.

CHAD: I wanted to make sure anyone with the skills could replicate what I'd done. If it wasn't open, it would have just been that metaphorical half-working project that probably would have been ignored by everyone (including me, eventually).

SIMON: Archival and preservation only works if it's open. I firmly believe that the only way to really preserve this type of history is the 'many copies, many mediums' approach. The unsung hero of any successful open project is documentation, though. Chad and I placed this high on the importance list, ensuring that the barrier for entry was low for newcomers.

LXF: Can you tell us a little more about the community you have with the Domesday86 project?

IAN: The community has expanded from just a few interested parties in the early days to a few hundred people actively using *ld-decode* and *vhs-decode*. There are hobbyists experimenting with the technology through to professional television engineers. The community has been an invaluable resource, As more people have discovered items and Googled what they are, they get in touch often to find out more and even donate items.

SIMON: The Domesday86 project and *ld-decode* are sister projects. On the Domesday side there has been a lot of community interest and assistance; people have provided valuable insight into the project and we've had

Photos can be opened from the Community Disc's map view.

Domesday86 team has developed SmallyMouse2, which provides a set of firmware, circuit diagrams, enclosure design and hardware to take a USB mouse as input and convert it to a protocol called quadrature, which was used by various devices, such as the BBC Master, Commodore Amiga and the Atari ST.

A budding community has built around the Domesday86 team, which is likely helped by the fact that the team has decided to develop code, hardware and enclosure designs in the open. Everything that is possible to be developed under an open source or other open licence has been. Once concepts have been understood, the information has been documented on the website, leading to a greater chance that the project will not be lost to history. Not only are the concepts explained, but they are then developed in code form, which is also open. Volunteers are then able to build their own SmallyMouse2 converters, Domesday Duplicators or run and make changes to the *ld-decode* software. It's no surprise that volunteers feel welcome in the community when they can see exactly what is happening and are able to make suggestions and changes themselves.

See for yourself

Is there a way to view this content? Well, thankfully, yes there is. The National Museum of Computing at Bletchley Park, UK, has working equipment on display for the public to use, as does the Centre of Computing History, which is based in Cambridge. The Cambridge museum also has an emulated version on its website *(see Quick Tip, page 71)*.

If you're not able to visit a museum that has a Domesday system, the best way for you to experience the Domesday Project is to open your favourite web browser and navigate over to **https://domesday.computinghistory.org.uk/**. This online exhibit was created with support from the Domesday86 project and a modified version of the open source *BeebEm* emulator, which was written in C++ and compiled to WebAssembly. When the page loads, select which disc you wish to view and wait for the page to load. Either open the Community North or Community South disc and use the keyboard shortcuts to navigate to your local area. If you're particularly lucky, you may even find information that was written by children from your own school.

The sheer power and flexibility of Linux, which can be off-putting to newbies, is what has made this project as approachable as it can be. Having all of the necessary programming languages, libraries and ancillary tools at one's fingertips allows for the very difficult task to be tackled.

We trust you've enjoyed this 'and now for something completely different' moment. The Domesday Project was ahead of its time, with its attempt to gather information from hundreds of thousands of children, collate it and store images, videos and textual data on one storage system, which is displayed and searchable from the BBC Master system and associated hardware. The interview section makes fascinating reading, especially when the team talks about the circular dependence on each other to improve things. If you wish to learn more, please visit the excellent Centre for Computing History or National Museum of Computing and ask as many questions as you can. Also try to use the lessons learned by the Domesday team and archivists when deciding what formats you would like to use for your projects. LXF

QUICK TIP

VHS-decode is a fork of *ld-decode* that can decode VHS tapes. It is also possible to sample tapes with a video recorder and *Domesday Duplicator*. See **https://github.com/oyvindln/vhs-decode** for more information.

QUICK TIP

More can be learned about the Domesday86 project at **www.domesday86.com**.

contact with nearly all of the primary project members from the BBC and beyond. The Domesday86 website is packed with info about the system and its history.

LXF: The geeky part: how many rooms' worth of Domesday-related hardware, LaserDiscs and other equipment do you have to enable you to do your work?

IAN: I've never put it all in one place and am scared to find out. I have been known to drive more than 150 miles with a LaserDisc player and scanners in the boot of the car to scan a rare disc or collect some paperwork.

CHAD: A bit all over the place. My main setup is in the corner of the living room with my main PC setup, and then I've got LaserDiscs in various places that I need to capture – or cull in the case of common stuff.

SIMON: Two rooms (I can hear my better half grumbling in the background as I write this). I have one room that is full of all my retro gear and another that contains all my software and hardware development equipment.

LXF: What about Linux makes your life a lot easier when working on Domesday or *ld-decode*-related matters?

IAN: My first real introduction to Linux was in writing the software to get analogue RGB frame captures from the real Domesday player under serial port control from Python. I could not coax the serial port library to work under Windows, which resulted in my first dual boot, and I haven't looked back.

SIMON: Linux is core to any modern open source development; there is no point in producing open hardware or software if you are then forced to use a closed environment to run it. As the project stabilised, pre-built ports for other operating systems have arrived, but I still prefer Linux – the available tools, especially around development, are second to none.

Stuart Burns is a Linux administrator for a Fortune 500 company specialising in Linux.

» ORANGE VS RED

As night follows day, a new kernel release has dropped. The 6.5 kernel seems to be general fixes for various bugs (for the x86/64 platform). There is also word that, separately, there is going to be a new version of the Vulkan graphics drivers for Intel Ark-based GPUs that teases at significant speed increases.

In the non-technical world, you'll be glad to read that the encrypted message intercepts submitted in the UK Online Safety Bill weren't codified into law. The government says they won't be made law until technically feasible, so will more than likely be looking to resurrect this in the future. Obviously, the policy makers don't understand that encryption is everything or nothing!

Within the corporate world, interesting things are afoot. The RHEL closed-source bun fight is bearing fruit for Red Hat. Except, it's not. I have been told by people in the know that Oracle will no longer be promising one-to-one bug compatibility. That means OEL will be doing its own thing post 9.2, but it did promise to try to maintain compatibility.

I'm also hearing that a lot of companies are abandoning RHEL/OEL for Ubuntu. Mostly due to cost and it being stable enough to run production workloads. That is not to say Ubuntu is better, just that the costs are far less, even if some of the support tools around large estate management are rather lacking. RHEL used to be the platform to build for but there is now a feeling, post-RHEL temper tantrum, that there is a lot more sense in using Ubuntu as a tested reference platform.

Red Hat says no more Screen time for you!

With the end of Screen in sight, it's time to use Tmux. Our man in the bowels of the server room explains.

Red Hat and a few other distributions have stopped shipping the much-beloved *Screen* utility. It's effectively a software console that you can run in the background. *Tmux* is not as easy as *Screen,* but moving forward, it's is the new standard on certain distros. *Tmux*, short for terminal multiplexer, is a powerful command-line tool that facilitates multiple sessions within one terminal window. It's especially useful for remote work, scripting and command-line multitasking, so here's a concise guide to using *Tmux*.

Start by entering `tmux` to initiate a session. Inside, you'll observe a green status bar indicating the session's active state. Want to return to your regular terminal momentarily? Press `Ctrl+b d` to detach.

Re-enter a detached session with `tmux attach`. For organised multitasking, name your sessions using `tmux new-session -s sessionname`. Reconnect using `tmux attach -t sessionname`.

Think of windows as tabs in a browser. Create one using `Ctrl+b c`. Navigate through them with `Ctrl+b n` (next) or `Ctrl+b p` (previous). For a split view within a window, use panes. `Ctrl+b %` splits vertically, while `Ctrl+b "` does so horizontally. Navigate between panes with `Ctrl+b` followed by an arrow key. *Tmux* transforms command-line usage, making multitasking seamless and efficient. Dive in, explore its features, and elevate your terminal experience.

Oi, stop Tmuxing around at the back, you lot, and pay attention!

Beware of geeks bearing gifts

We're still trying to work out why you need a download manager anyway...

Linux malware isn't something any careful Linux user encounters very often. That makes it all the more shocking when it comes to light that there has been a Linux application in plain sight that wasn't all that it appeared.

Freedownloadmanager.org was offering a download tool for Linux users. However, behind the scenes, a trojan was offering its authors full access to the user's machine. It also functioned as a data stealer, grabbing anything useful it found, including cloud provider credentials.

The most startling thing about this was the fact that this trojaned software was first seen in 2020 but no one picked up on it. More information on this specific story can be found at **https://bit.ly/45XeHJZ**.

While an interesting tidbit, it should also serve as a warning: when looking at software, it is worth taking the time to consider what the software is and where it is coming from. Staying inside the well-established repos means the chances of something untoward happening is vanishingly low.

Going outside the trusted and established repositories should be treated as a last resort. Consider where you are getting the files – can they be trusted? At the same time, we all know it's easy to make package managers ignore GPG checking and other sanity checks, but it really isn't worth it.

CREDIT: https://github.com/tmux/tmux/wiki/

Seek and ye shall locate

How to find files in Linux using Find, Locate and Grep...

Searching for files in a command-line Linux environment can sometimes be a non-trivial task, especially when dealing with numerous directories and large data sets.

Being able to perform these tasks via the Linux command line pays massive dividends, however, and makes you look good when you have mastered search. It also means that it is easy to automate the task and make it repeatable.

It's important to remember that Linux is all about using the right tool for the right job well, and chaining tools together to achieve an outcome.

The powerful tools that can assist users in finding files include *Find*, *Locate*, and *Grep*. Let's explore the basics and how to chain them together with pipes.

Finders keepers

Find is a versatile tool that searches for files and directories in a hierarchy based on given criteria. It's particularly useful when you're uncertain about a file's exact location. To find all TXT files in the current directory and subdirectories is quite straightforward:

```
$ find . -name "*.txt"
```

Useful, but not awesome. Additional parameters extend it. Below is a useful example of the **-type** and **-size** parameters. As an example, to find all files larger than 10MB in the **/home/user** directory, use:

```
$ find /home/sysadmin -type f -size +1M
```

Obviously, it goes without saying that you can use the redirect **>** and **>>** to output the console text to a file. An example is shown below:

```
$ find .txt -type f -size +1M > results.txt 2> /dev/null
```

Using the above puts all the files into **results.txt** and sends any errors to **/dev/null**.

Locate and behold!

While *Find* searches through the filesystem in real time, *Locate* uses an index, making it faster, though potentially out of date. We admit this is one super-useful and speedy way to use files and is one of our go-to ways to locate files. To find all instances of a file called **notes.txt**, use:

```
$ locate notes.txt
```

Again, some switches can be used to fine-tune the output. To implement a case-insensitive search for a directory named **Projects**, use:

```
$ locate -i projects
```

Pro tip: if you are looking for a file that matches a specific directory and filename, you can use:

```
$ locate bin/postgres
```

This would report only where there was a directory called **bin** that contained a file called **postgres**.

The *Locate* command relies on there being an up-to-date database. By default, it reindexes once every 24 hours. It is possible to manually rerun the indexing by using `sudo updatedb`. It should take very little time on most systems.

Grep expectations

While *Find* and *Locate* search for files based on metadata, *Grep* searches within files for a specific pattern. It's a text search utility that can identify lines in files that match a given pattern:

```
$ grep [options] pattern [file...]
```

An example of this is using it to search for the word 'Linux' in a file called **document.txt**:

```
$ grep "Linux" document.txt
```

Most admins use case-insensitive search to make sure all instances are picked up (unless it's specific capitalisation) for the word 'Linux' in all TXT files within the current directory:

```
$ grep -i "Linux" *.txt
```

Grep can also be used to recursively search for the word 'Warning' in all files, starting from the current directory:

```
$ grep -r "Warning" .
```

Piping hot commands

These commands are great, but by combining them with others, it lifts the searching power substantially. Pipes, represented by the `|` symbol, enable you to take the output of one command and use it as input for another. This makes it easy to combine the capabilities of different commands for more complex tasks.

Going back to *Find,* we can use the pipe command to say that any log files that are larger than 10MB should have something done to them:

```
$ grep -r "Warning" . | tail -n 5
```

Using a pipe to push the output to *Tail* means that only the last five warnings are shown, rather than every warning in that folder.

As another example, if we want to archive those files found, it is possible to use the **-exec** parameter to do so. By default, the `gzip` command also removes the original, so proceed with caution.

```
$ find  -type f -name "*.log" -size +10MB -exec gzip {};
```

Note the new **-exec** switch – it is a way to run a command on the output. To search for files smaller than 10MB, use `-` to specify 'less than', so `-10MB` could be used. *Find* also understands MB, GB and KB.

Linux provides a rich set of commands to aid in the task of file searching. Whether you're looking for a file based on its metadata with *Find* or *Locate*, or you're searching for a specific pattern within a file using *Grep*, Linux has you covered. Additionally, by harnessing the power of pipes, you can combine these commands in various ways to achieve your desired result.

Mastering these commands is crucial for anyone aiming to be proficient in Linux, because they'll undoubtedly prove indispensable in your daily tasks. LXF

NordVPN

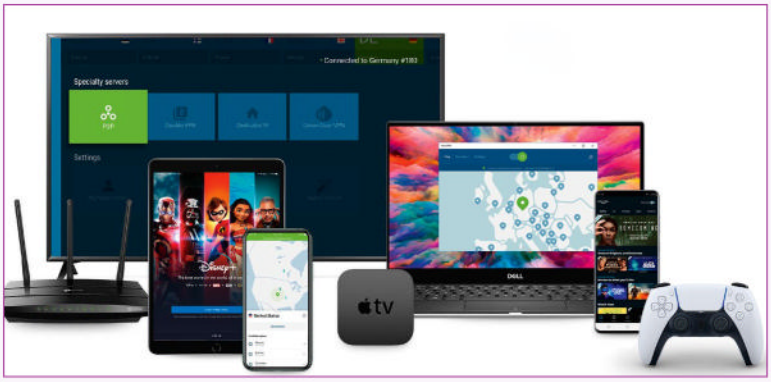
Mike Williams puts one of the big-name VPNs under his microscope.

IN BRIEF

NordVPN is an appealing VPN provider with top-of-the-range performance, loads of features, and a reassuring no-logging audit. It's not the cheapest, and we have a few small issues with the app interfaces, but the company has added a bunch of welcome improvements recently, and overall it's a polished and professional service that delivers good results for most users.

Panama-registered NordVPN is a hugely popular VPN provider with more than 14 million customers. The company sells itself on features, and there are plenty to explore.

The NordVPN network has 5,600-plus servers in 85 locations across 59 countries. We ran tests on several locations to confirm that its servers are physically located in the advertised countries, and it was good news all around. From Costa Rica to Estonia, UAE to Vietnam, every server we checked was in its promised location.



NordVPN offers support for a huge range of OSes, devices and services. Even Linux.

CREDIT: NordVPN

You get Windows, Mac, iOS, Android and Android TV apps, a feature-packed (if command-line) Linux offering, and tutorials to set up the service on many more platforms and device types. Plus *Chrome, Firefox* and Edge plugin extensions.

NordVPN boasts support for six simultaneous connections. That means you can set it up on as many devices as you like, but only six can be connected at once. This is probably enough for most people, but if you have a lot of hardware to protect, IPVanish, Private Internet Access, Surfshark and Windscribe have no set limits on connections or devices.

Essentials and extras

NordVPN scores well on the VPN essentials, with OpenVPN and its own WireGuard-based NordLynx protocol providing strong encryption, a rock-solid kill switch and DNS leak protection to avoid data leaks.

Powerful extras include Double VPN and Onion support for extra security. There are P2P-optimised servers in most locations, and an audited no-logging policy gives strong reassurance that your internet activities won't be visible to anyone else.

NordVPN includes ad and tracker blocking, its Meshnet enables you to create a secure encrypted network with up to 10 of your devices, and up to 50 others (as long as they're also NordVPN users). This could allow you to share files, play network games, access other network devices and more.

If you're intimidated by this feature overload, or just run into some unexpected problems, NordVPN's 24/7 support teams are on-hand via email or live chat.

NordVPN uses strong AES-256-GCM encryption and supports perfect forward secrecy to regularly change key, ensuring that even if an attacker manages to penetrate one session, they're locked out of the next. Once you're connected, it uses its own private DNS to keep your browsing away from third parties.

NordVPN claims to have a strict no-logs policy. Most VPNs say the same, but the company's privacy policy clearly defines this. It covers not only general logging of internet activities, but also session logging details such as recording your incoming IP address when you connect to the service, and the IP you're allocated.

To back this up, NordVPN hired Deloitte to run an independent audit on its infrastructure and services, and to verify that its logging policy description is accurate. In January 2023, Deloitte's concluded that it "saw no signs that we in any way violated our no-logs promises". While some VPNs still boast about a single audit they took five years ago, this is NordVPN's third for its no-logging policies alone.

NordVPN's own NordLynx protocol delivered spectacular download speeds of over 950Mb/s. That puts it equal first in our speed charts along with Surfshark, IPVanish, Hide.me, Norton and Windscribe.

OpenVPN is slower, but useful if you're setting up NordVPN on a router – our tests managed 240Mb/s. We've seen faster (Mullvad reached over 500Mb/s), but that's more than enough for most situations.

NordVPN doesn't explicitly claim to unblock any particular website or service, but our test proved successful with Netflix, Disney+, BBC iPlayer, ITV and Channel 4. It's a shame Linux support (while wide) only extends to a terminal-based command. LXF

VERDICT

DEVELOPER: NordVPN
WEB: www.nordvpn.com
PRICE: From £2.49 per month

FEATURES	8/10	EASE OF USE	7/10
PERFORMANCE	9/10	VALUE	8/10

Blistering speed, top-of-the-range unblocking and audited no-logging policy make for an appealing service. But there are minor usability issues and prices are dearer than most.

» Rating 8/10

Cargo

A website builder with a user-friendly interface and customisable features – **Ruby P Jane** is suitably impressed.

IN BRIEF

Cargo's website builder excels in its user-friendly interface and customisable features, making it a solid choice for creating sites. Although its performance and pricing face competition, which may be a setback for some users, its ease of use and creative options shine, setting it apart.

Cargo is a website building service that provides a platform for creating and designing websites with a focus on visual aesthetics and user-friendly functionality. Its interface is designed to facilitate the smooth arrangement of content, including images, text, multimedia and interactive elements, enabling you to showcase your work, products or ideas in an engaging and visually appealing manner. Through its intuitive drag-and-drop interface and customisable templates, Cargo enables you to craft unique and dynamic websites that reflect your creative ideas.

Design and layout customisation is at the core of Cargo's offerings, granting you the freedom to experiment with various templates, fonts, colour schemes and styles. This versatility ensures that every website maintains a distinct visual identity, catering to diverse tastes and purposes.

The integration of image galleries within Cargo facilitates the creation of picture galleries and videos. You can effortlessly curate and display your portfolios or showcase products through interactive slideshows, enhancing engagement and aesthetic appeal.

For anyone venturing into online selling of goods and services, Cargo also offers ecommerce functionality. This enables you to showcase and sell your products or services directly through your website, streamlining the purchasing process for potential customers and maximising revenue potential.

Stack attack

The stacking and pinning features provide precision in content arrangement, granting you the ability to layer elements to create dynamic and visually engaging layouts. Stacking enables the overlay of different elements, giving you a sense of depth and interaction, while pinning ensures certain elements remain fixed during scrolling, maintaining their visibility and impact.

Cargo's backdrop feature adds another layer of customisation, enabling you to apply backgrounds that harmonise with your content. This ensures a consistent and immersive visual experience throughout the site.

The service also offers a range of customisable templates, each designed with various industries and creative niches in mind. These templates provide you with a solid foundation to build upon, enabling you to easily tailor your websites to match your unique style, brand or content. You can customise fonts, colours, images and more with straightforward controls, ensuring that the final design aligns with your creative vision. The menus on the dashboard are also well spelt and easy to find and navigate.

The initial pricing tier, the first plan, stands at a reasonable $14 (£11) per month. However, for those looking to go into ecommerce, the plan starts at $19.50 per month. This plan is tailored to accommodate the ecommerce features of the platform, providing a range of tools to facilitate online selling.

Cargo is offering a style all of its own.

If you want more features, the second plan, priced at $19 per month, opens the door to a wide range of enhanced functionalities. The ecommerce features of the second plan cost $28 per month.

There is also a free tier for education, with its student sites offering. For students looking to showcase their work, projects or ideas, this specialised offering enables them to make use of the power of the platform completely free of charge.

Cargo's website builder stands out as a user-friendly tool for creating captivating websites without complex coding. With a focus on visual appeal and creative freedom, it empowers individuals, artists and businesses to showcase their work online.

Although its performance and pricing are competitive, its ease of use and customisable features are where it shines. Against rivals such as Wix and Squarespace, Cargo's unique strengths make it a solid choice for those seeking an impressive online presence. In a world where a strong digital identity matters, Cargo offers an accessible and effective platform to bring imaginative visions to life on the web. LXF

VERDICT

DEVELOPER: Cargo
WEB: http://cargo.site
PRICE: From $14 (£11) per month, payable yearly

FEATURES	**6/10**	**EASE OF USE**	**9/10**
PERFORMANCE	**9/10**	**VALUE**	**6/10**

Cargo's website builder excels in its user-friendly interface and customisable features, making it a solid choice.

» Rating 7/10

CREDIT: Cargo

Seventh level of networking

Understanding networking needn't be hell. **Darien Graham-Smith** takes us through the ISO OSI model that runs it all!

Networking is one of those magical areas of technology where complex processes are made beautifully simple for the end user. Operations such as downloading a web page or backing up a file to a NAS appliance take just a few clicks or taps. Behind the scenes, though, there's a lot of highly advanced engineering going on.

Do you need to know about the under-the-bonnet technical details? Perhaps not. But if you have an understanding of what's going on, physically and digitally, that can help you troubleshoot networking problems when they arise, and write your own programs that make efficient use of networked resources. It's interesting, too – and the OSI model makes it highly accessible, providing an easy way to visualise the processes that makes a network tick.

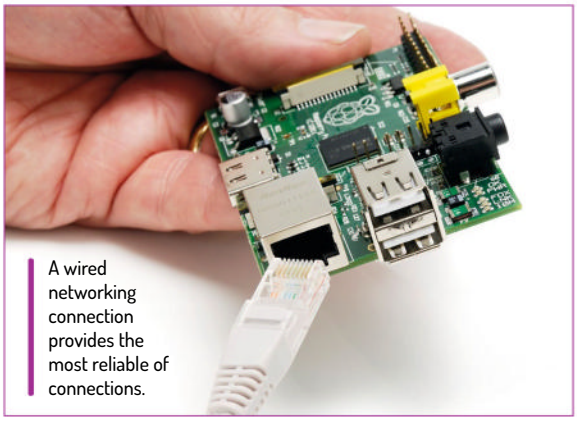
What is OSI?

The OSI model is a structured description of the operation of a computer network. Its name identifies the body that created it, namely the Open Systems Interconnection group. That group is part of the International Standards Organisation, so if you're palindromically inclined, you could call it the ISO OSI.

The OSI model was developed in the '70s and '80s to provide a standard way of talking about networks that could aid in designing, building and managing every level of a functional network, from the physical wires right up to how individual programs and services communicate with one another. The OSI model is intentionally couched in general terms, focusing on functions rather than specific technologies – so it's still wholly applicable to modern networking systems.

The most distinctive feature of the OSI model is that it breaks networking down into seven layers. These are hierarchical, in the sense that each one receives some sort of input from the one below (except the bottom layer), and processes it in some way for the benefit of the one above (except, obviously, the top layer). If there's a problem with any individual layer, the layers above it probably won't work properly.

Although the OSI model is an international standard, it could be argued as a British innovation. Its origins lie in a 1976 paper by the UK's National Computing Centre that called for the creation of such a standard, and its early development was supported by the UK's Department of Trade and Industry, plus researchers at British universities. But the first formal specification was published in the US in 1978 by French engineer Hubert Zimmermann, and the defining multi-layered approach was built on earlier work by US computer scientist Charles Bachmann at Honeywell Information Systems in the '70s. We can share the credit.



A wired networking connection provides the most reliable of connections.

Ethernet cabling has its own CAT standard to support 100MB, Gigabit, 2.5 Gigabit and the latest CAT7 for 10 Gigabit Ethernet.

Layer 1: Physical

When the OSI model was created, the foundation of all networking was the cable itself. That makes sense: without some sort of connection between devices, no aspect of the network can function. So, the lowest layer of networking is the physical layer.

The physical layer doesn't just refer to cables running between network nodes, however. It also covers the adherence to agreed standards that make communication possible over those cables.

For example, a Gigabit Ethernet card needs to provide an RJ45 socket that's engineered to specific, defined dimensions, and it needs to have electrical contacts in a particular arrangement, which can send and receive electric pulses of a defined voltage and duration. The cable needs to have a compatible plug, with the right number of internal strands of wire, and


7. Application layer
Protocols for programs to talk to one another
7
6. Presentation layer
Encoding and encrypting data for transport
6
5. Session layer
Keeping connections open across multiple operations
5
4. Transport layer
Assembling packets into usable data
4
3. Network layer
Routing packets to the right device
3
2. Data link layer
Translating electrical signals into binary code
2
1. Physical layer
Electrical signals between computers
1

Dumb hubs just repeat all incoming network traffic to all ports, hence a switch uses packet switching to forward network traffic to an actual single destination.

Powerline adaptors have to translate the Layer 2 from standard Ethernet cable transmission to over electrical lines.

appropriate conductivity and shielding to carry the signals reliably over the specified length – typically 100 metres for a Cat5e or Cat 6 cable. To be clear, though, Layer 1 isn't limited to conventional Ethernet; it covers all forms of connection, such as fibre optic links and modern wireless tech such as Wi-Fi and Bluetooth.

At this lowest level, the details of what's actually sent over the connection are irrelevant. Layer 1 is about signals – as long as the desired patterns of voltages, or light, or radio waves are being successfully conveyed between stations, Layer 1 networking is working.

Layer 2: Data link

Layer 2 deals with translating Layer 1 signals into meaningful communications. That means converting those incoming electrical impulses – or whatever they may be – into ones and zeros (and doing the same in reverse for outgoing connections).

Importantly, Layer 2 doesn't process the incoming data. Its role is solely to extract coherent digital messages – known as frames – which can be passed to processes in higher layers to read and work on.

However, Layer 2 isn't completely disconnected from the content of the data stream. Its role also includes monitoring the incoming signals in order to recognise when a remote machine is starting to send it data and when it's finished. It typically also inspects the data frames themselves to check their integrity. If an incoming frame appears to be corrupt, it's a Layer 2 function to ask the originating computer to resend it or to pass an error condition up to a higher-layer process.

Layer 3: Network

The name of this layer may seem rather broad, considering the topic at hand. You can alternatively think of this as the addressing layer or routing layer – that is, the layer that deals with ensuring that network communications reach their intended destination.

In most home and office settings, addressing is handled by giving every network node a unique IP address (or at least one that's unique in that network). Frames are then wrapped up in data packets that include the destination address at the start of every transmission, along with their payload of binary data.

Where the source and destination computers are both connected to a router or switch, that network device normally forwards outbound packets directly to the client with the specified address. But Layer 3 also needs to deal with destinations that aren't directly connected, such as those on different subnets or out on the internet. These may only be reachable by passing the data packets on to other routers or clients, which forward them in turn. You can see how TCP/IP packets travel through one or more 'hops' to their destination by opening a command prompt window and typing (for example) `mtr ubuntu.com` or `mtr 192.168.0.1`. This open-ended route-finding and packet forwarding is the essence of Layer 3.

Unlike Layers 1 and 2, Layer 3 functions aren't necessarily the responsibility of any individual device. They require the networking components of all devices and operating systems to work together; if data packets aren't reliably travelling from any client to any other, the network as a whole has a Layer 3 problem.

Layer 4: Transport

This is another name that might seem counterintuitive – you'd be forgiven for assuming that transport referred to the cabling. In fact, the transport layer does in a sense provide a low-level service, but it does so at the application level. After the lower three layers have taken care of turning electrical signals into bits and delivering them to the correct computer, Layer 4 is responsible for delivering them to a particular app (or operating system component) in a usable form.

This isn't quite as simple as copying a block of binary data from one buffer to another. Data normally travels over a network in a series of chunks, known as datagrams. Breaking up the data like this helps the free flow of network traffic, and reduces the amount of information that has to be resent if a datagram is lost or corrupted. However, it also means that the chunks need to be reassembled at the destination computer, and potentially resequenced to account for delayed or retransmitted datagrams. All this happens in Layer 4.

Layer 4 also provides a second order of error checking. While Layer 2 functions verify that individual frames aren't corrupted, Layer 4 works on a larger scale to confirm there aren't any chunks missing. If Layer 4 is doing its job, the destination system receives a stream of data over the network in much the same way as if it were reading a file from the local hard disk.

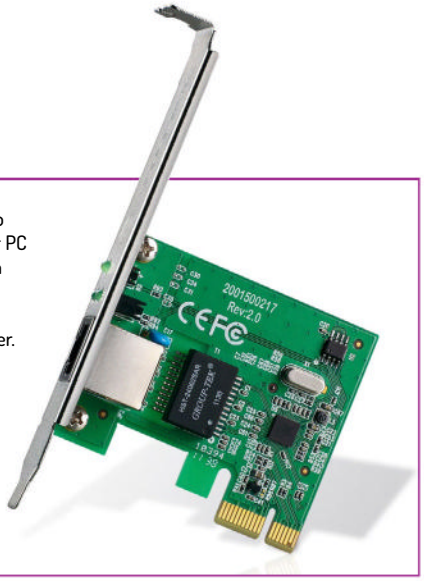
Layer 5: Session

While the lower four OSI layers focus on individual network operations, Layer 5 deals with longer-lasting sessions – open-ended communication channels that remain available until they're intentionally closed.

A simple example would be the connection between a client PC and a file server. Once you've logged into

QUICK TIP

A good example of Layers in action is with the Wake-on-LAN (WoL) packet. This is actually a Layer 2 protocol, it's directed to a specific MAC address and can't be routed via TCP that's Layer 3. It also means you can't send a WOL over the internet to your home device. Some routers however do offer a proxy for this.



If you want to upgrade your PC to faster than Gigabit, an add-in PCIe card is in order.

the server, you can browse, copy files and do whatever else you want, until you log out. At that point, the session is over and you can't access the server again without logging back in. Applications can use the same model to talk directly to one another over the network without user interaction.

Behind the scenes, the connection may not really be held open, but opened and closed as needed. Even so, the concept is useful, and widely used. Working in persistent sessions is convenient for humans, and ideal for encrypted communications. Rather than having to establish a new secure channel for every data exchange – which would be very inefficient – systems can generate a per-session encryption key and use it for the duration of a particular set of interactions.

Layer 6: Presentation

Talking of encryption, that's one of the services provided at Layer 6, the presentation layer. This layer includes all functions that encode data into a desired form for transmission, and decode incoming data.

Obviously, this doesn't mean duplicating the Layer 1 task of converting zeros and ones into electrical impulses. Rather, it means applying mathematical transformations, such as encryption or compression, to increase the security or efficiency of the transmission. Any encoding applied by a Layer 6 function is expected to be decoded by the corresponding function at the other end, so its effect should be invisible to the end user (unless they're monitoring the raw traffic flowing over the network).

Indeed, Layer 6 is also responsible for ensuring interoperability between clients, typically by converting character codes to universally readable alternatives, or to whatever is expected by the receiving computer. For this reason, it's sometimes called the syntax layer or the translation layer.

Layer 7: Application

The application layer sits at the top of the stack. This comprises the software interfaces and protocols that apps use to interact directly with one another, such as HTTP for web browsing or SMTP for email delivery.

It's important to understand that the applications themselves are not part of this layer. Indeed, they're not included in the networking structure at all. Once a web page or email has been handed off from Layer 7 into an application, that data has left the networking stack and become local to that application.

The same applies in reverse. When you type an email into *Thunderbird,* you're working with local data. When you hit Send, the application copies the details of your message into a request to a Layer 7 networking component. This feeds it through Layer 6 functions and so forth until the message reaches Layer 1 and is sent out into the world in the form of a series of electrical pulses or radio waves. The local copy remains and is perhaps saved in your **Sent Mail** folder.

This might seem like a nit-picking distinction, but the point of the OSI model is to break the complex process of networking down into atomic elements. If you can't send an email, it's valuable to be able to distinguish whether the problem is with the application, or with SMTP itself.

Indeed, for most of us, that's the main benefit of the seven-layered approach. When a networking operation goes wrong, understanding how the communications process breaks down into different layers can help you narrow down where the problem is likely to lie. LXF

» THE FOUR-LAYERED TCP/IP STANDARD

The seven OSI layers don't directly mirror any particular networking system, and that's arguably a strength. It means the model can be used to analyse or plan out any sort of networking scenario.

However, not everyone agrees that OSI is a good fit for the 21st century. That's because the vast majority of modern networks use TCP/IP – and this particular collection of standards is defined in terms of four layers, which don't perfectly match the OSI divisions.

In TCP/IP, the bottom layer is the link layer, or network access layer. This includes defined standards for both connection protocols (such as Ethernet and Wi-Fi) and basic communications over those connections, so it's like a combination of OSI Layers 1 and 2.

The next two TCP/IP layers are the internet layer and the transport layer. These cover much the same ground as OSI Layers 3 and 4 respectively. The name internet layer isn't exclusively for internet connections, but refers to the general ability to route packets between different LANs – something the OSI model includes in its more loosely named network layer.

The major divergence from the OSI model is that TCP/IP doesn't recognise session and presentation management as separate layers. Its fourth and top layer is the application layer, which may optionally include software routines that handle encryption, character encoding and other such matters.

With only four layers, the TCP/IP model is simpler to work with. And because the layers relate directly to published specifications and standards, it is more practical for troubleshooting and network architecting.

Even so, the OSI model is almost 50 years old, and its broad applicability means it's not going anywhere soon. When network engineers talk about "Layer 2 load balancing" or "Layer 3 switch hardware", they're using the OSI definitions. They're universal terms that are understood across networks of all types, and will probably still be long after TCP/IP has fallen by the wayside.

HotPicks

Xournal++ » Parallel Disk Usage » Mission Center » Floorp » Ferdium » HopToDesk » Touché » Sonic Robo Blast 2 » Fheroes2 » Ulauncher » AlbumEasy

Mayank Sharma
hasn't paid attention to Red Hat since the CentOS saga. Besides, all he needs is his trusty Ubuntu box to test-run his HotPicks.

HANDWRITTEN NOTES

Xournal++

Version: 1.2.0
Web: https://xournalpp.github.io

If you like scribbling notes, but can't be bothered to move away from the computer to find a pen and piece of paper, you'll like *Xournal++*. Besides keyboard and mouse, the app also supports digital pens, and in addition to writing notes, it can also be used to annotate PDFs.

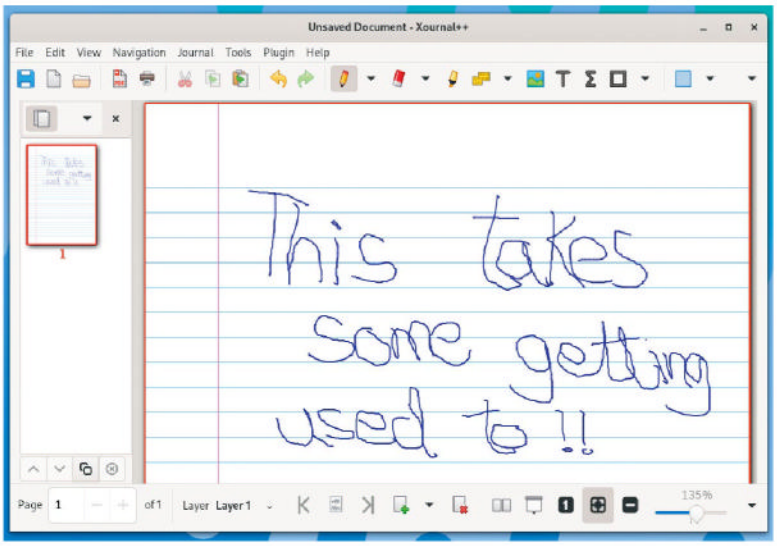
It's available in the repos of all the popular desktop distros, including Fedora, Ubuntu, Debian, OpenSUSE and their ilk, and can be installed from your distro's default package manager. The project also produces distro-agnostic Snap, AppImage and Flatpak binaries, which don't just simplify the installation, but also ensure you have the latest release of the app.

The app has a busy interface that might confuse first-time users, but should quickly grow on you. After launching the app for the first time, head to Edit > Preferences, and switch to the Input System tab. There you can ensure the app has detected all the input devices connected to your computer and assigned them the correct role.

One of the main benefits of *Xournal++* is its flexibility. You can, for instance, handwrite notes using the mouse, keyboard, touchpad or touchscreen, as well as pressure-sensitive styluses and tablets, from a wide variety of manufacturers including Wacom, Huion, XP-Pen and more.

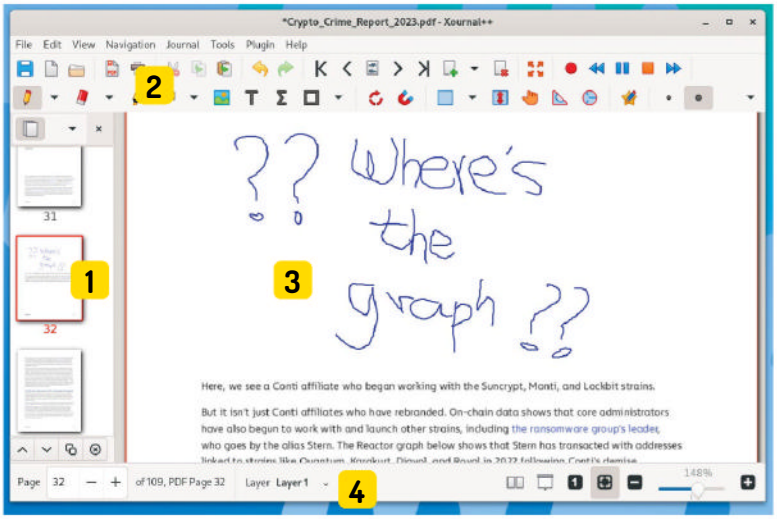
To write a note, select the Pen tool from the toolbar. You can change stroke attributes, such as its shape and colour, using the toggle buttons and drop-down menus in the toolbar. To annotate a PDF, use the File > Annotate PDF option to select the relevant PDF file. You can then use tools such as the Highlighter tool, Pen tool and Text tool to add elements on top of the PDF's pages.

Once you've created a note or annotated a PDF, you can export your creation to a variety of formats including SVG, PNG and PDF. First-timers should refer to the detailed user guide on the *Xournal++* website to make the most of the app.



The latest release of Xournal++ has been under development for two years, and enhances several of the app's existing functionalities.

LET'S EXPLORE XOURNAL++



1 Sidebar
Enables you to preview both the pages and the layers in the current document. You can use the buttons at the bottom of the sidebar to create new pages.

2 Toolbars
Spread over multiple rows, the toolbars display all the tools that you can use to create and modify elements that can be used for adding annotations.

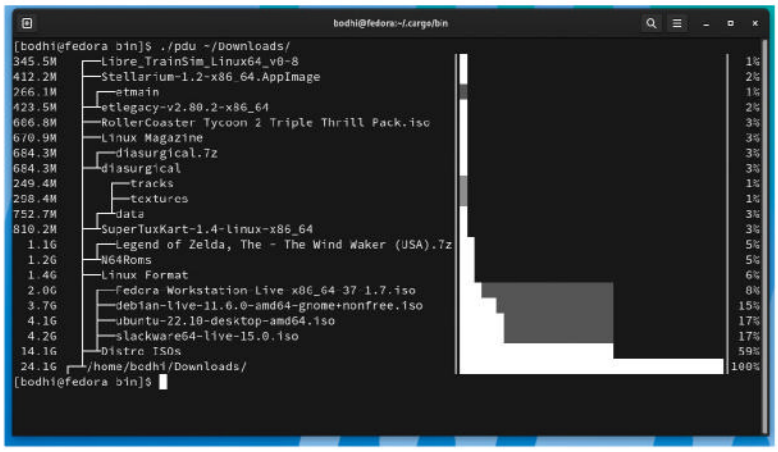
3 Canvas
This is where the app works its magic. It displays the currently loaded journal or PDF, and enables you to scribble your annotations.

4 Bottom menu
Has useful controls such as the ability to zoom in and out of a document, and to navigate the layers of a page using the layer selection box.

DISK ANALYSER

Parallel Disk Usage

Version: 0.9.0 **Web:** https://crates.io/crates/parallel-disk-usage



Like all good CLI utilities, PDU offers options for everything from customising its output chart to limiting trawling depth, and a lot more.

Linux has no dearth of tools for reporting disk usage. One of the most frequently used routes is to restrict the output of *Du* using its plethora of options, before piping the output into a variety of other tools, such as *Sort*.

While this works for small directories, on much larger ones it doesn't just take a considerable amount of time, but without the right options, also produces illegible output. That's where *Parallel Disk Usage (PDU)* steps in. Not only is it a lot faster than plain old *Du*, but it also produces visually appealing output.

PDU's written in Rust and can be installed with its *Cargo* package manager. To install *Cargo,* you'll first have to install Rust with `curl https://sh.rustup.rs -sSf | sh`. This downloads a script and starts the installation. When it's done, it automatically installs *Cargo*. Once you have *Cargo,* you can use it to install *PDU* with `cargo install parallel-disk-usage --bin pdu`.

The `pdu` command displays the disk usage chart of the current directory. You can also specify the path of a directory, such as `pdu ~/Downloads` to view its usage.

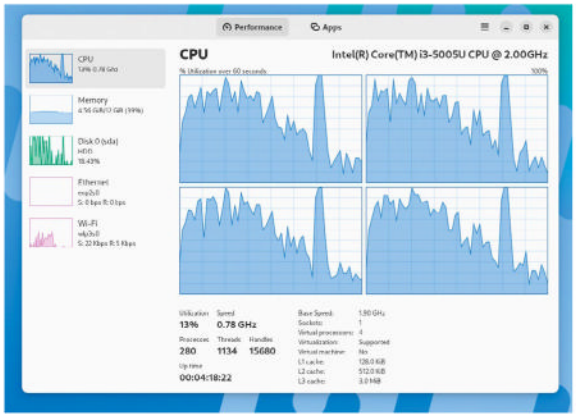
PDU displays all the files and directories beneath your chosen destination, along with their relationships, their size and an incredibly useful bar chart that gives you a quick overview of which files and folders are taking the most space. And it does all this in the blink of an eye, irrespective of the size of the directory.

By default, the chart shows a percentage value for how much of the destination space a specific file or directory is taking up. For a CLI app, this is a wonderful way to get a better understanding of the disk usage, and can help you find hidden cache directories that might have ballooned to incredible sizes.

SYSTEM MONITOR

Mission Center

Version: 0.3.1
Web: https://missioncenter.io



When a process uses large amounts of CPU or RAM, Mission Center highlights it under the Apps tab.

System admins go to great lengths to keep an eye on the resource utilisation on the systems in their realm. But it's always a good idea even for desktop users to do the same to ensure all is hunky-dory with their computer. Every desktop distro ships with a task manager that usually does a good job, but some still leave room for improvement.

Mission Center is a dedicated system monitor and task manager that does an excellent job of helping you keep track of the apps and processes running on your installation, and how they are using resources.

It's available as a Flatpak and can be installed with `flatpak install flathub io.missioncenter.MissionCenter`.

You can use the app to monitor the overall CPU, memory, disk and network usage. In addition to overall CPU usage, you can also view usage per logical processor. Additionally, the app also displays lots of useful information including the number of running processes and threads, system uptime, maximum and current processor clock speed, cache sizes and more.

Similarly, when it comes to memory usage, in addition to RAM and swap usage, the app also displays info about the RAM sticks, including their form factor, type and speed. You can also use the app to monitor disk use, transfer speeds and several other details.

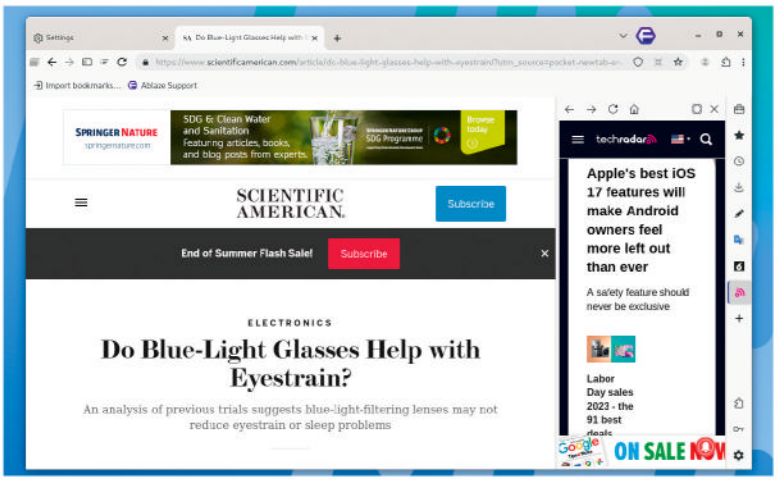
For network monitoring, the app has separate displays for Ethernet and Wi-Fi, and displays the name of the chipset on both, along with their hardware and IP addresses, as well as connection-specific details such as wireless speeds and frequency for Wi-Fi.

The app also displays a list of current active apps and processes, along with their individual CPU, memory and disk usage. On the downside, you can't use the app to stop, kill or end the processes.

WEB BROWSER

Floorp

Version: 11.1.2
Web: https://floorp.app



Firefox is certainly a popular web browser, but that doesn't mean it can't be improved, and *Floorp* attempts just that. Based on the popular open source browser, *Floorp* markets itself as a feature-rich flexible alternative.

It is available as a portable app on Linux, and also offers a PPA for Ubuntu users. The most convenient option is to use the distro-agnostic Flatpak release – install it with `flatpak install flathub one.ablaze.floorp`.

On first launch it does appear quite similar to *Firefox* with a few UI differences. For starters, you can import your bookmarks, passwords and more from an existing *Firefox* installation by logging into your *Firefox* account.

That said, the browser, based on the *Firefox ESR* release, has stripped the *Firefox* telemetry code.

In terms of security, *Floorp* offers pretty much the same protections that you get with *Firefox*, including automatic tracker and third-party cookie blocking, an HTTPS-only mode, pop-up blocking, and the like.

Floorp also bundles the uBlock Origin content filtering extension by default. And with feature updates every four weeks, its developers claim they can provide updates and fix security issues faster than *Firefox*.

The biggest difference in the interface is *Floorp's* built-in sidebar that you can use to access bookmarks, history, downloads and more. The app prides itself on its extensive layout customisation options. Within the settings section, you can move the position of the tab bar to the bottom of the window or hide the title bar.

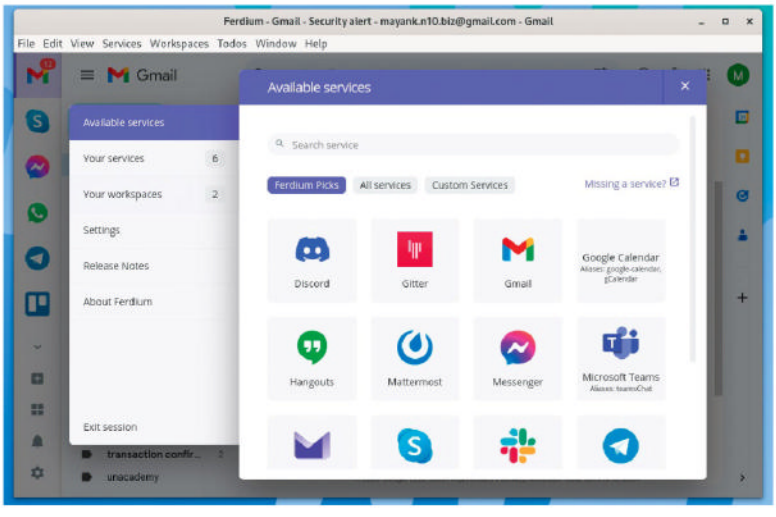
You can also add multiple tab rows, switch tabs by scrolling, navigate the browser using mouse gestures, customise a website's appearance, and even tweak the picture-in-picture video controls. Similarly, you can customise *Firefox's* usual themes and switch the browser's interface between multiple different designs.

Floorp claims you can, in essence, use its numerous customisation options to create your own instance of the browser.

UNIFIED WEB SERVICES

Ferdium

Version: 6.4.1
Web: https://ferdium.org



The idea of a unified messaging app isn't new, but widen the app's mandate to cover not just messaging services, but all kinds of web apps, and you've got a winner. *Ferdium* does just that thanks to its support for over 200 web apps and services.

Ferdium puts out both RPM and DEB binaries, but is also available as distro-agnostic Flatpak, AppImage and Snap packages. Use `flatpak install flathub org.ferdium.Ferdium` to install it from FlatHub.

When launched, the app gives you the option to create a free account. This ensures you can access all your configured services across all your *Ferdium* installations across different computers.

For the privacy conscious, there's also the ability to use the app without creating an account, but you lose the ability to cloud sync your account info with other computers because your app data stays with you.

Getting *Ferdium* to handle your services is a two-step process. First you browse through the list of supported services and find the one you want to add. By default, text and audio notifications are enabled for all added services, although you can disable them while adding the service. You can also change its name – for instance, home email, work email and such – and even use a custom icon. Then, once you've added a service, give it your login credentials and you're good to go.

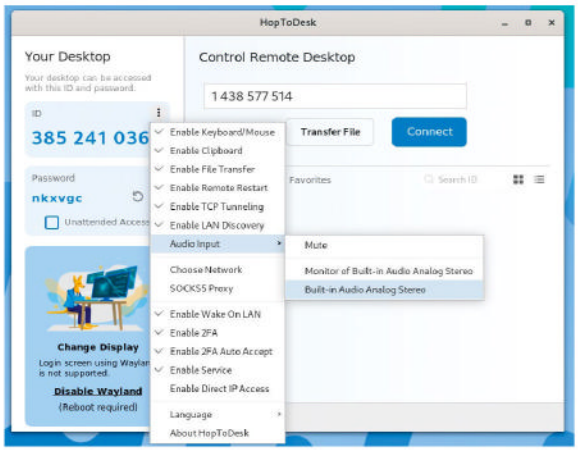
One of the good things about *Ferdium* is its configuration options, using which you can tweak everything from the app's appearance to its behaviour. One of the interesting settings is its hibernation policy, which helps save resources by unloading unused services after a specified amount of time. Paired with its wakeup policy, this setting ensures that you don't miss any updates, while also saving resources.

Ferdium also enables you to group configured services into workspaces – for instance, to separate your personal services from your work-related ones.

REMOTE DESKTOP

HopToDesk

Version: 1.40.5
Web: www.hoptodesk.com



In addition to end-to-end encryption, HopToDesk also lets you toggle two-factor authentication for additional security.

There are tons of very competent and popular remote desktop apps. Most of them make it remarkably easy to share your screen, as well as remote control and access other devices.

HopToDesk is one such application: a remote desktop tool that's designed to provide lots of useful features without compromising on security and privacy.

Unlike other similar tools, such as *TeamViewer* or *AnyDesk, HopToDesk* is free for both personal and business use, and is available for all popular desktop and mobile operating systems.

The project puts out binaries for popular distros including Ubuntu, Fedora, OpenSUSE and others, and is also available as a distro-agnostic AppImage. Just grab the AppImage, make it an executable either from the file manager or with `chmod +x`, and double-click on it to launch the app.

Using the app is reasonably straightforward. When you fire it up, it spits out an ID for the machine it's running on, along with a password. Enter these details on the *HopToDesk* instance running on another computer or device, and you can remotely control the original computer.

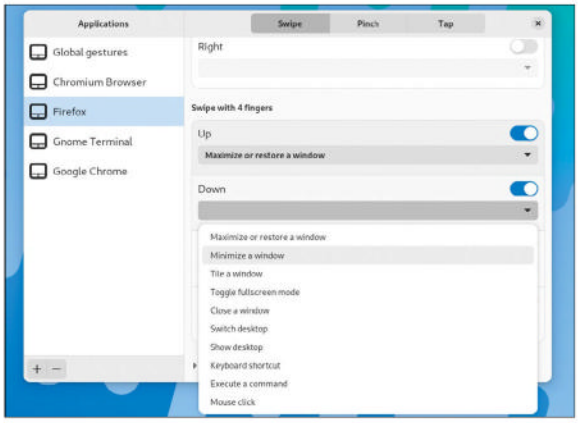
In addition to screen sharing and remote control, you can use *HopToDesk* to transfer files between the connected computers, as well as exchange instant messages. In terms of security, all traffic, including chats, screen sharing and file transfers, between the devices is protected with end-to-end encryption.

Furthermore, while all the features are enabled by default, the app does give you the choice to selectively turn off certain options that you don't want to expose. For this, use the three-dots menu next to ID to bring up the list of included features. From here, you can disable any of the options that you don't want, such as file transfer or remote restart.

GESTURE CONFIGURATOR

Touché

Version: 2.0.9 **Web:** https://github.com/JoseExposito/touche



While the majority of gestures on Touché work on touchpads, certain gestures listed towards the end work only on touchscreens.

If you use a laptop or touchscreen, you can increase your productivity by installing *Touché*. It's a multi-touch gesture configurator that helps you use the touchpad more efficiently.

Touché is basically a graphical wrapper around the *Touchégg* utility, which runs in the background and does all the heavy lifting. So, before you install *Touché* you need to grab *Touchégg*.

The good thing is it's probably already installed on your distro if you use recent versions of Linux Mint, Pop!_OS, Elementary OS or Zorin OS. It is also available in the official repos of some distros, such as Fedora, and can be installed with `sudo dnf install touchegg`.

Ubuntu users must grab it via its official PPA:

```
$ sudo add-apt-repository ppa:touchegg/stable
$ sudo apt update
$ sudo apt install touchegg
```

Next, grab *Touché* as a Flatpak with `flatpak install flathub com.github.joseexposito.touche`.

While you can use *Touché* on any desktop environment, if you're using the Gnome desktop, you should also install the X11 Gestures Gnome Shell extension (**https://github.com/JoseExposito/gnome-shell-extension-x11gestures**).

The app has a straightforward interface. At the top you have three tabs to control the gestures for swipe, pinch and taps. Note, however, that tap gestures are only available on touchscreens. On the left you have the option to define these gestures globally, as well as per individual application.

By default, *Touché* lists a couple of apps, but you can use the + button to add any open application.

Defining the gestures is pretty straightforward as well. By default, the app enables certain gestures. To define a custom action, scroll down the list of gestures and use the pull-down menu to select from one of the predefined actions.

SONIC CLONE

Sonic Robo Blast 2

Version: 2.2.11
Web: www.srb2.org

SRB2 lets you relive the original, and playing the game in 3D makes it even more fun and challenging.

Sonic the Hedgehog was a mainstay of '90s console gaming. *Sonic Robo Blast 2 (SRB2)* combines *Sonic* with another behemoth of the era: *Doom*. *SRB2* is built using a modified version of the *Doom Legacy* port of *Doom*, and has been in development since 1998.

The developers don't distribute binaries for Linux, and instead point to the Flatpak created by its community that you can install with `flatpak install flathub org.srb2.SRB2`.

The game features several levels and the gameplay is similar to the original, but has been boosted to three dimensions. Another addition in *SRB2* is the multiplayer mode that enables players to either play together as a team or against each other.

The game has several multiplayer modes, including Coop, which is the same as single-player mode but with multiple players, Competition, Capture the Flag, and more. The multiplayer modes can be played either via split-screen or over a network connection.

You get to play as Sonic, Tails or Knuckles, each with their own abilities, but all three characters can spin dash, a move that's distinctive of *Sonic*.

Your mission remains the same: collect as many rings as possible while traversing each zone as quickly as you can. On the way, you tackle many of the same enemies and hurdles as the original.

Currently, *SRB2* features six full-length zones. The short first one, Greenflower Zone, is meant to ease players into the game. *SRB2* also has a Record Attack mode, where you can replay the levels to get a high score, by either collecting more rings or completing the levels faster.

As *SRB2* is built on top of the *Doom* engine, it is easily customisable, and the multitude of user-made add-ons is one of the reasons for its popularity.

STRATEGY

Fheroes2

Version: 1.0.7 **Web:** https://ihhub.github.io/fheroes2

Fheroes2 doesn't muddle the gameplay of the original, but enhances it by piling on features found in modern strategy games.

Another classic from the '90s, *Heroes of Might and Magic 2* was one of the most popular fantasy turn-based strategy games of the era. *Fheroes2* is a recreation of the game that has been written from scratch and promises to deliver the charm of the original along with improvements to the gameplay, graphics and logic.

The game is available in the official repos of popular distros such as Fedora, but the developers recommend you grab the game from Flathub with `flatpak install flathub io.github.ihhub.Fheroes2`.

To play the game, you need a copy of the original. When you launch *Fheroes2*, it gives you three options. The recommended one is to grab the original from GOG. If you have the original on a CD, you can go for the second option and point *Fheroes2* to the original's files, following the instruction's on the game's website. Or, you can opt for the third option and let *Fheroes2* download a demo copy of the 1996 classic.

With *Fheroes2*, you can start a standard game, where you can select a map and decide on the character you want to play, or you can play the game with others in multiplayer mode.

One of the enhancements is a battle-only mode. *Fheroes2* also gives you the option to finish battles automatically, along with the ability to replay a battle manually if you're not happy with the results.

Experienced campaigners will be able to spot changes and enhancements throughout the game. For instance, the game displays the special characteristics for each creature through the creature info window, which also displays the duration and description of spells.

All said and done, if you have played the original, you'll appreciate the enhancements *Fheroes2* brings to the table, which should also make the classic appealing to those used to modern strategy games.

APP LAUNCHER

Ulauncher

Version: 5.15.3
Web: https://ulauncher.io

The search bar in the Applications menu in modern desktops is pretty useful for finding apps and documents. *Ulauncher* does all that and more, and is a whole lot more configurable than your built-in application launcher.

It is available in the official repo of Fedora, and can be installed with `sudo dnf install ulauncher`. Ubuntu users can install it through a PPA with:

```
$ sudo add-apt-repository ppa:agornostal/ulauncher
$ sudo apt update
$ sudo apt install ulauncher
```

Launch *Ulauncher* – if your desktop has a system tray, you'll see the app's icon in there. Even if your desktop doesn't have a system tray, you can press Ctrl+Space to invoke the *Ulauncher* search bar.

You can start typing and *Ulauncher* runs through all of your installed apps to find a match. The good thing about the app is that it uses fuzzy search, which means you can misspell the name of the application and still be able to get to it.

The app also has a couple of built-in shortcuts to find more than just applications. Use `~` or `/` in the *Ulauncher* search bar to turn it into a file browser. Similarly, type `g`, `so` or `wiki` to search for the succeeding text on Google, StackOverflow and Wikipedia, respectively.

To define your own custom shortcuts and to add more functionality, head to the app's Preferences section by clicking the gears icon in the *Ulauncher* search bar.

To add more functionality through an extension, head to the Extensions tab and select Discover Extensions. This takes you to the app's extensions website. Browse through the list and when you find an interesting extension, copy its URL and paste it into the Add Extension text box in *Ulauncher*'s Preferences section.

You can also change the default hotkey and even make Ulauncher start automatically when you log in to the desktop.

STAMP ALBUM LAYOUTS

AlbumEasy

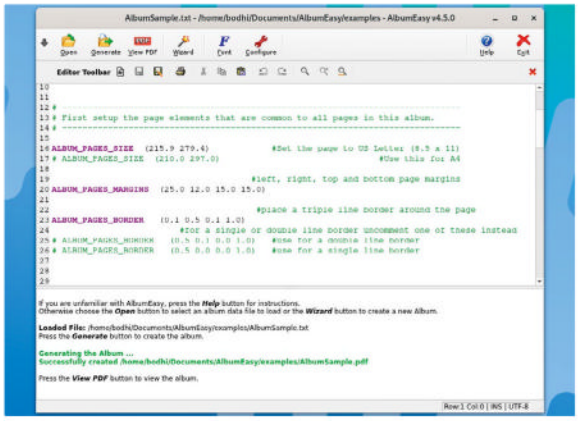
Version: 4.5.0 **Web:** www.thestampweb.com/albumeasy

If you ask us, the most popular software in the world is open source. And that holds true even on the other end of the spectrum, when it comes to niche software. And they don't get any more niche than *AlbumEasy*. If you like to collect stamps, you can use *AlbumEasy* to print custom pages to stick them on.

To install the software, grab its tarball and extract it, then move the extracted folder to an appropriate location, such as **~/AlbumEasy**. Now change into this folder and launch the application from the CLI with `./AlbumEasy.sh`.

If your desktop can display desktop icons, you can also run the **config** script with `./CONFIG`, which creates a desktop shortcut for the app and also moves the **examples** folder inside the **~/Documents/AlbumEasy** folder. This folder contains several example templates for custom pages.

AlbumEasy works by reading a text file that contains a description of the album pages, which it then parses to generate a PDF file that you can print.



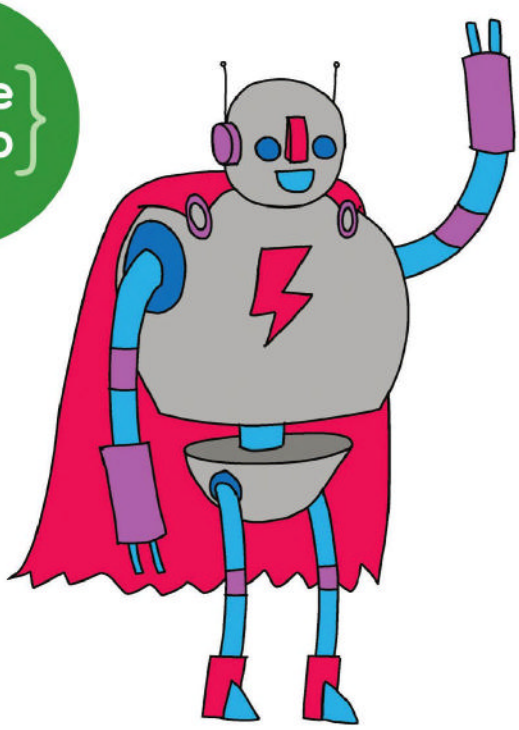
The app has a minimal interface. You can create an album definition file from scratch, but it's a good idea to modify one of the existing ones in the example folder and tailor it to your needs.

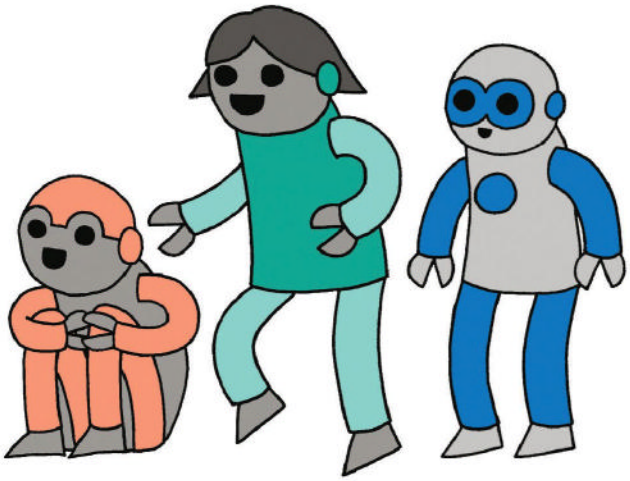
Click Open, navigate to the **examples** folder, and select the **AlbumSample.txt** file. As you can tell, an *AlbumEasy* template is essentially a standard text file that contains a list of app-specific commands.

The **AlbumSample.txt** file does a nice job of explaining some of the most-used commands. You can also refer to the project's thorough documentation on its website for a detailed list of all supported commands and their functions. There are also several video tutorials to help you get started. Once you've created your description file, smash the Generate button to create the PDF. LXF

You can create an AlbumEasy definition file in a standard text editor and then compile it into a PDF from the CLI.

code club





Can you help inspire the next generation of coders?

Code Club is a nationwide network of volunteer-led after school clubs for children aged 9-11.

We're always looking for people with coding skills to volunteer to run a club at their local primary school, library or community centre for an hour a week.

You can team up with colleagues, a teacher will be there to support you and we provide all the materials you'll need to help get children excited about digital making.

There are loads of ways to get involved!

So to find out more, join us at **www.codeclub.org.uk**

HOME ASSISTANT

Credit: www.home-assistant.io

Part One! Don't miss next issue, subscribe on page 16!

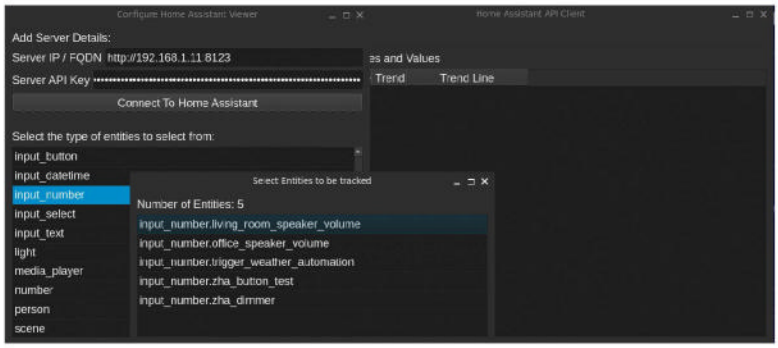
Visualise smart-home sensor data

Matt Holder investigates how to take data from an API and display it in a home-built interface.

OUR EXPERT

Matt Holder has worked in IT support for over a decade, and always tries to utilise Linux alongside the other installed systems.

Over the past decade or so, the computing world has changed immensely and we're going to capitalise on one particular area to grab data from a source and display it on the screen in a fancy Dan interface. It's a useful ability to have because you can quickly prototype displaying important information from a range of sources when needed.



We are creating a GUI with three windows that reads data from the Home Assistant API.

We're going to work on an application that pulls data from the *Home Assistant* API and displays it in a GUI. This GUI enables the entities that are shown to be selected by the user and displayed in a table, and the values to be updated at regular intervals. As an added bonus feature, we are implementing a way to display whether the value has stayed the same, increased or decreased since the last reading.

Home Assistant is an amazing project and provides the glue that allows many different home automation tools to be used together. For example, data from a Nest thermostat could be used to switch on and off smart plugs, rather than just your boiler. This is a tiny example of what *Home Assistant* can do. It consists of lots of entities that store values about smart-home objects around your home. These entity values can be returned from the API. The API is capable of more, but that's beyond the scope of this article.

Whether you are a *Home Automation* user or not, however, the concepts discussed in this tutorial will enable you to undertake similar projects that match your own interests.

This project is split into two areas. We're first creating the necessary classes to read data from our API and then we're creating the GUI functionality, which we use to draw the necessary widgets on the screen, provide the GUI interactions code and call the API code to extract the necessary data. Splitting the program in this manner is sensible as it means that the API library could easily be reused for a different project, such as making a command-line program or widget for your favourite desktop environment.

So, let's first introduce the application program interface, or API as it's known to its friends. APIs provide access to data that can then be used for other purposes. It may be that the data is simply written to a file or database, displayed in a GUI, used to submit to a different API, or a combination of some or all of the previous examples. APIs can be unauthenticated or locked behind authentication, and this point is important as you probably wouldn't want a complete stranger reading your up-to-date location using the API from *Home Assistant*.

So, how does querying an API actually work? Well, put simply, our Python code submits a request to a server via http (or more likely https) with some authentication details and perhaps some payload information. As with usual http requests, there are multiple types available. The **GET** type is generally used when requesting information and the **POST** type is used to request changes to be made. For example, if we wish to request the value of a sensor, we would use **GET** and if we wish to set the state of an entity or fire an event, we would use a **POST** request.

Any information returned from the *Home Assistant* API is in JSON format, which is incredibly common,

QUICK TIP

The complete code listing can be downloaded from https://github.com/mattmole/LXF308-309-API-GUI.

although XML may be used as well. Our code contains some methodology to try to convert values into native data types. Any states we return from *Home Assistant* are strings within the JSON extract and we'll try to convert these to numbers so that we can process them as such.

The *Home Assistant* API contains a number of different endpoints. We are using the one that lists all states by name and the one that returns the state of a given entity. The first endpoint is used to return all states and this is used to populate two locations on the program's GUI. Once the user has selected which entities should be displayed on the GUI, we then query those particular ones and display the latest value on the screen.

So, without further ado, let's set up the environment we're using for our coding. Python has a system called virtual environments that can be used to perform coding in a more sandboxed environment. Once created, this venv, as it is known, contains its own set of installed libraries that the rest of the system cannot use. This helps with making sure it is as simple as possible to separate a particular project from another one. Open your terminal and carry out the following commands (this is based on a Debian-derivative distribution):

```
$ sudo apt install python3-pip
$ sudo apt install virtualenv
$ mkdir Documents/Programming/LXF308 -p
$ cd Documents/Programming/LXF308
$ virtualenv .
$ source bin/activate
```

What we're doing here is installing the necessary tools to be able to create the virtual environment and then creating one. We create the directory the code will live in, change to it and create a new venv. Once created, using the source command allows the venv to be activated. Now, any commands that are run are contained within the venv, rather than on the host filesystem. Once the source command has run, the prompt changes so that you know where you are working. Next, we can install the necessary Python libraries that we are going to be using. Back in the terminal, enter the following commands:

```
$ pip install PyQt6
$ pip install PyQt-tools
$ pip install rich
$ pip install matlotlib
```

Now this has been created, open your favourite IDE and create a file within this new directory called **haApiClient.py**. The first thing we now do is import the libraries we need to pull the data from the *Home Assistant* API:

```
#! /usr/bin/python3
from requests import get
import json
from urllib.parse import urljoin
from rich import print
import time
import configparser
import os
```

In this next sample of code, we are creating a class that interacts directly with the API and returns the raw data. Following the code listing, we'll describe what the operation does.

```
class HaApiClient:
  def __init__(self,uri="",apiKey=""):
    self.uri = uri
    self.headers = {}
    self.headers["Authorization"] = f"Bearer {apiKey}"
    self.headers["content-type"] = "application/json"
    self.data = {}
    self.response = {}
    self.getStatesEndpoint = 'api/states'
    self.responseCode = None
    self.responseJson = None
  def getRequest(self,endpoint):
    response = get(endpoint, headers=self.headers)
    self.response = response
    self.responseCode = response.status_code
    if response.status_code >= 200 and response.
status_code < 400:
      self.responseJson = json.loads(response.text)
  def getStates(self):
```

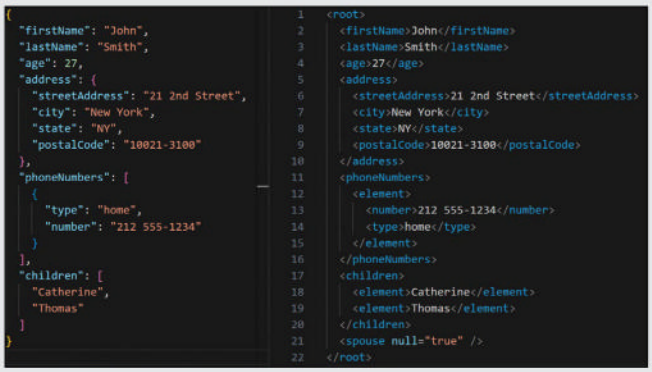
QUICK TIP

You can learn more about the *Home Assistant* API at **https://developers.home-assistant.io/docs/api/rest/**.

» JSON VERSUS XML

What is JSON? It stands for JavaScript Object Notation and is an open standard that is used for data exchange and file storage. JSON is human readable and uses key: value format to attribute a value with a key. It is hierarchical, too, and while the name contains JavaScript, it is widely used in a huge number of programming languages. JSON can represent single values as strings and can also store lists of items as well.

XML stands for eXtended Markup Language and is a markup language and file format for storing and transferring data. As can be seen, the intended purpose is the same as with JSON. When looking at the example *(right)*, it is clear to see that JSON uses far less text to store the same information and that both are fairly clear in how the data is stored. The biggest difference in XML is how the items within a list are represented. XML nodes, as they are called, can also contain attributes, such as `<firstName atBirth="true">John</firstName>`, which can represent if John was the user's name at birth or has been changed since. Using this example from Wikipedia, you can see how JSON compares with XML in the screenshot.



Comparing JSON (left) and XML (right) shows how much lighter JSON is.

QUICK TIP

To give you ideas for your own projects, this list contains details of a huge number of public APIs: https://github.com/public-apis/public-apis

```
    endpoint = '/'.join([self.uri,self.getStatesEndpoint])
    self.getRequest(endpoint)
  def returnStates(self):
    self.getStates()
    return self.responseCode, self.responseJson
  def getState(self,entity_id):
    endpoint = "/".join([self.uri, self.getStatesEndpoint,
entity_id])
    self.getRequest(endpoint)
  def returnState(self, entity_id):
    self.getState(entity_id)
    return self.responseCode, self.responseJson
```

The first thing we have completed in this sample of code is to create a new class called **HaApiClient**. The **__init__** function is used to initialise the class and this code is run whenever a new instance of the class is created. The function takes the URI of the server and API key as arguments. The **init** code then adds these details to variables. The **self.headers** variable is used to store the authentication details. The **self** keyword refers to the variable as belonging to the instance of the class. We then utilise variables to be able to store data returned from the API, response code, JSON from the API and the endpoint address to be able to pull states from the API.

The second function we define is **getRequest**. This pulls data from the API. The **get** method from the requests library is used to query an endpoint address and the **self.headers** variable is used to send the authentication data to the API. Once the data has been returned, the response and status code are stored in variables. Assuming a successful connection, the JSON returned from the API is stored in another variable.

We next define the **getStates** function, which takes an API endpoint address as the argument. Within the function we combine the base URI with the endpoint address and then call the **getRequest** function.

The **returnStates** function is pretty simple and is used to call the **getStates** function and then return a tuple of the response code and any JSON data. This function is used to return all entities from the API. The **getState** function is similar to **getStates**, but this pulls state information for individual entities. Finally, **returnState** formats the data returned by **getState** and lets the user use it.

```
class HaEntityStatus():
  entities = {}
  entitiesList = []
  def __init__(self,uri,apiKey, entity_id = ""):
    self.uri = uri
    self.apiKey = apiKey
    self.entity = entity_id
    self.returnCode = 0
  def readAllEntities(self):
    apiCall = HaApiClient(uri = self.uri, apiKey = self.
apiKey)
    entities = apiCall.returnStates()
    self.responseCode = apiCall.responseCode
    if entities[0] == 200 or entities[0] == 201:
      HaEntityStatus.entities = entities[1]
      #Now return a list of just the entity IDs
      for entity in entities[1]:
        HaEntityStatus.entitiesList.
append(entity["entity_id"])
  def readEntity(self, entity_id = ""):
    if self.entity == "":
      self.entity = entity_id
    #Check to see if the entity_id exists in the list of
entities
    if self.entity in HaEntityStatus.entitiesList:
      apiCall = HaApiClient(uri = self.uri, apiKey = self.
apiKey)
      entity = apiCall.returnState(self.entity)
      if entity[0] == 200 or entity[0] == 201:
        return {"responseCode":entity[0],
"responseJson": entity[1]}
      if entity[0] > 201:
        return {"responseCode":entity[0],
"responseJson": {}}
    else:
      print("Entity does not exist")
```

In this code sample, we create a second class, which we use for formatting the data that is returned from the API. We can also be clever about its usage and store a copy of all entities into what is called a class variable – this is a variable that is shared by all instances of a class. In our case, this is not strictly necessary because we do not need to use the list of entities multiple times. However, if we were to need this list, it would speed matters to store a copy of all entities that all instances can reference, rather than pull the entity list for each instance. This code should not be used with multithreading because it is pretty unlikely that the code is thread-safe, but it does demonstrate an important principle.

Once the class has been defined, we create two class variables called **entities** and **entitiesList**. One of these is a dictionary and the other is a list. We then

» UTILISING OTHER FUNCTIONALITY FROM THE API

The *Home Assistant* API is capable of doing a lot more than simply returning values. *Home Assistant* integrates with a huge number of systems and allows data to be stored centrally, as entities. As well as this, automations can be created that take data from one system and provide an action on another. For example, if a PIR detects movement, it'll then switch on a light. Another is that a service can be called based on values from a different system.

The *Home Assistant* API is capable of returning a list of events and services as well as entities. Events are back-end messages and the event bus is central to everything that *Home Assistant* does. Returning this information is only half of the story, though, and by using **POST** methods, it is possible to add data to the event bus, fire a service or set the value of an entity.

A useful example of this would be to have what is known as a helper that could be set by clicking a button in our application. This could then fire an automation to perform some tasks. Clicking a button when starting a voice call could trigger an automation to switch on a 'do not disturb' light and switch off your smart speaker from playing music. The only limit is your imagination.

An example of data from Home Assistant can be seen here. Temperature data being stored can be returned and used in different ways.

initialise the class with the **__init__** function, which takes the server URI and API key as well as the **entityId** that we wish to query. We then use some instance variables (**self.varName**) to store the information that we pass into the **__init__** function and some variables to store the results.

The **readAllEntities** function is used to call the **HaApiClient** methods to query the API and return all states. If the response code indicates success, we add the API output to the **entities** class variable as well as adding each **entityId** to the **entitiesList** class variable. Class variables are referenced using the **className.varName** syntax rather than the **self.varName** syntax.

The **readEntity** function is similarly used to query the API via the **HaApiClient returnState** method and if the data is returned successfully, it is formatted as a dictionary, which contains a key and value for the http response code and one to return the JSON that relates to the entity being queried.

With this code written, running it does not accomplish anything. What we have done here is define classes, which contain the necessary logic to query the API and return the data. What we have to do next is create instances of our classes, pass in the API details and then call the necessary functions. The code listing on GitHub contains a sample command-line client, which requests numbers that correspond with **entityIds** and displays them every 10 seconds. The code below simply prints all entities returned from the API.

```
if __name__ == "__main__":
    uri = "http://IP_ADDRESS:PORT"
    apiKey = "API KEY"
    allEntities = HaEntityStatus(uri, apiKey)
    allEntities.readAllEntities()
    entitiesList = allEntities.entitiesList
# Sort the returned data
entitiesList = sorted(entitiesList)
print(entitiesList)
```

In this code sample, we first of all use an **if** statement, which only runs the code if the code is run directly. If we were using the classes from another Python program, this code would not run. First of all, we define variables to store the server URI and API key. Next, we create an instance of the **HaEntityStatus** class, using the URI and API Key as arguments, and store the results in a variable called **allEntities**. Next we call the **readAllEntities** method to return all entities from the API. On the next line we take the list that contains all of the **entityIds** and store them in a variable. Finally we sort the list into alphabetical order and then print it to the screen.

We trust that you've enjoyed this first part of our tutorial series and that it has given you an idea of how to integrate with third-party systems by utilising their APIs. Next month, we will write a GUI and use it to display the information. LXF

```
1485 sensor.all_standby_power
1486 sensor.all_standby_energy
1490 sensor.tx
1491 sensor.tx_2
1492 sensor.rx
1493 sensor.rx_2
1494 sensor.tx_total
1495 sensor.tx_total_2
1496 sensor.rx_total
1497 sensor.rx_total_2
1502 sensor.backup_state
Enter entity id. (Enter q to finish entering list to check) 735
Enter entity id. (Enter q to finish entering list to check) 89
Enter entity id. (Enter q to finish entering list to check) q
[735, 89]
200 Playroom Sensor - Temperature 23.0
200 CPU Temperature 72.1
200 Playroom Sensor - Temperature 23.0
200 CPU Temperature 72.1
200 Playroom Sensor - Temperature 23.0
200 CPU Temperature 72.1
```

A simple command line client can be found here and is included in the code at GitHub.

LAZARUS

Credit: www.lazarus-ide.org

Write GUI apps in Pascal with Lazarus

David Bolton explains how to get started with Lazarus and Free Pascal to develop GUI applications running on Linux.

OUR EXPERT

David Bolton has 30 years of Pascal programming experience under his belt, so knows a thing or two about developing GUI programming applications on Linux and other platforms.

Creating GUI apps in Linux is more complex than making simple apps that run in a terminal. There are several ways that are mostly *GTK*-based with C or C++, such as *GTK+*/C, *Qt*/C++, *FLTK*/C++ and *WxWidgets*/C++. You can also use the Python library PyGObject.

There is another, easier way if you don't mind learning Pascal. Install *Lazarus (see boxout opposite)*, an IDE that also installs the cross-platform open-source programming language Free Pascal. This lets you create GUI apps visually by dropping icons on a form, setting their properties, then compiling and running the program.

After installing it, you can start *Lazarus* with the command `startlazarus` in a terminal. That opens the *Lazarus* editor. You should see five different windows:

1. The top one is the main window, with the menu and all components and their icons.
2. To the left is the Object Inspector. This shows the properties for the selected component on the Visual form *(see 4)*.
3. The source editor is where you edit the source code of your Pascal application.
4. The gridded window with the title **Form1** is the Visual form.
5. A Messages window at the bottom. This shows the compile results.

GUI applications in Free Pascal are made up of several file types. There's the main program file, which has an LPR extension (short for Lazarus PRoject). It's not visible initially but click the Project menu and then click View Project Source (at the bottom of the menu that appears), and it opens in the source editor.

Anything with a `$` inside comment braces `{$}` is a compiler directive. The first directive `{$mode objfpc}` tells it to compile in objfpc mode (see the online documentation). The `{$H+}` tells it to use long strings.

The **uses** block is a list of units that are needed to provide functions and procedures. In Pascal, a unit is a module that is linked in to your application at compile time. It's a handy way of splitting your application into smaller parts. Free Pascal comes with many standard units, such as forms, classes, sysutils and interfaces. You can also create your own units.

The `{$IFDEF` compiler directives wrapped around the unit names are conditional directives. Like this:

```
{$IFDEF condition}
Unit name,
{$ENDIF}
```

The **cthreads** unit is only included if this is compiled on a Unix system. The **cthreads** unit links in POSIX thread routines. No prizes for guessing on what platform the **HasAmiga athreads** unit is used for. If your code will never run on an Amiga, feel free to delete those three lines.

The `{$R *.res}` tells the compiler to link in any RES files. These hold icons, strings and so on.

The `begin .. end.` is the equivalent of a `main()` block in C/C++. To see where **Application** is defined Ctrl+click on any instance of **Application** – it opens up the **Form** unit source code where **Application** is defined.

```
program example;

{$mode objfpc}{$H+}

uses
 {$IFDEF UNIX}
 cthreads,
 {$ENDIF}
 {$IFDEF HASAMIGA}
 athreads,
 {$ENDIF}
 Interfaces, // this includes the LCL widgetset
 Forms, Unit1
```

QUICK TIP

The source code for this example is zipped up in the file Lazarus-example.zip at https://github.com/David-H-Bolton/Projects. Just open the link in *Firefox*, click on the file, then the three dots on the right, and download it.

This is the top window, which has the main menu and tabs for all the visual components.

```
  {you can add units after this};

{$R *.res}

begin
  RequireDerivedFormResource:=True;
  Application.Scaled:=True;
  Application.Initialize;
  Application.CreateForm(TForm1, Form1);
  Application.Run;
end.
```

Structure of a Pascal program

The program starts with `Program` and the name. This name is used for the generated executable. Each form used in the program is a unit. By default, Lazarus starts with a simple one-form program. This form is created at run-time by the line `Application.CreateForm (TForm1, Form1);` in the main program file.

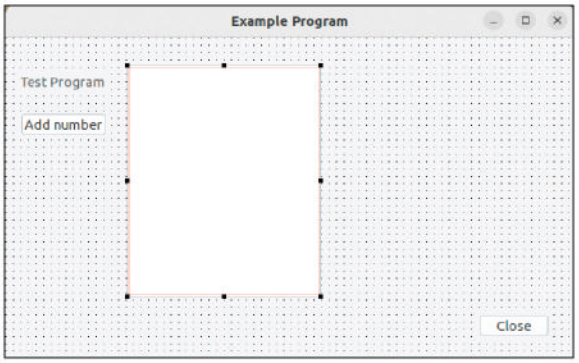
On the IDE top window, there's a row of tabs: Standard, Additional, Common Controls and so on. Each tab has many icons, though most of the time you probably only need those on the Standard and Additional tabs.

Each icon is a visual component that you can click on and then click on the form to drop it there. Ignore the first two icons, they're for menus, the third one is a button and the fourth one is a label. If you move the cursor slowly over each icon on the tabs, it shows its name. All component class names start with a T, so a button is of type **TButton**, list is **TListBox** and so on.

With the Form View selected, start by dragging the edge of the form to make it a bit taller and wider. Now click on a **Label** component (on the Standard tab on the top menu), then click on the form. This drops a label there. Next add a **Button** below it in the same way – click the button icon on Standard, then click the form. Add a **Button** in the bottom-right and a **ListBox** in the middle. It should look like the screenshot *(below)*, except the names will be different.

Click on the label – it will say Label1. Now look at the Object Inspector window. That's the tall window on the left. It shows a list of properties for whichever control you last clicked as well as a tree view of the controls at the top. You can click any component in the tree to select it in the Form View.

The Object Inspector after clicking on a label looks like the screenshot *(over page, bottom-left)*. There are two columns for the Properties – the name of the property and its value that you can change. Some properties are more complicated and clicking on the value shows a button that pops up a form to change that value. Try the **Font** property to see an example.



The Design View is where you can drop controls and drag them around.

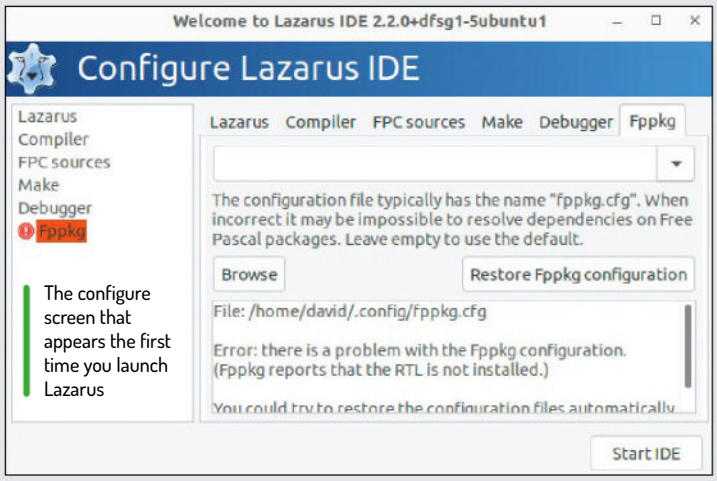
You can change the text by clicking on the property value for the **Caption** property and typing text in. We changed the **Name** property for several of the controls. Just scroll down the Object Inspector to view any properties that are off-screen. We used **btn** as a prefix for the buttons, so the Add Number button is **btnAddNumber**, the Close button is **btnClose**.

You can reposition controls by dragging them with the mouse or manually editing the **Top** and **Left** properties. To remove a control from a Form, just select it, then press the Delete button on the keyboard.

There are lots of properties, so have a play. Change colours, size, text, whatever you want. If you want tooltips to appear, set the Form's **ShowHint** property to **true**, then set the **Hint** property for each control.

The source editor.

This is a text editor that can handle multiple files. You use it mainly for editing PAS files but you can also view the text file of a Form file. Each Form is made up of two files. One has the details of the Form and its controls and is a text file with the extension LFM. The other has



The configure screen that appears the first time you launch Lazarus

» INSTALL LAZARUS AND FREE PASCAL

Installing *Lazarus* and Free Pascal is straightforward. Do the usual update/upgrade to get everything stable, then do the install:

```
$ sudo apt update -y
$ sudo apt upgrade -y
$ sudo apt install Lazarus
```

To launch *Lazarus*, in the terminal just type:

```
$ startlazarus
```

The first time it runs, it opens the Configure form. Work your way through the six tabs. You want to see OK in each one in the bottom text area. Only a couple of them need attention. You have to enter the path to *GDB* on the Debugger tab. You can find this by using the Files application – click the search box and enter `gdb`.

As there's no *Fppkg* config file, switch to the Fppkg tab then click the Restore FppKg configuration button. This opens another form that lets you generate new *Fppkg* configuration files. Just click the Write New Configuration Files button at the bottom. It returns you to the Configure screen and you can now click the Start IDE button. From then on, use `startlazarus` to open it straight into the IDE.

QUICK TIP

It's a good habit to rename components so you know what they do. For example, instead of button1, button2 and so on, give them meaningful names, such as btnAddNumber. This makes it easier to understand your code.

QUICK TIP

Free Pascal isn't just good for GUI apps. It can also create console apps, libraries, web browser applications, Node.js applications, *Atom* packages and *Visual Studio Code* extensions. Just click Project on the top menu, then New Project to see a form open with the various project types.

the source code for what the Form does and is a text file with a PAS extension. The Form View shows the .LFM as controls and the Code View shows the source code in the source code editor. F12 toggles between the two views. To view the Form file as a text file, just right-click on the Form, then click View Source (.lfm).

Objects in Pascal

In code, each form is a class subclassed from the class **TForm**. Classes are like C++ classes and are used to define objects. After the class **frfinition** you'll see a var then `Form1 : TForm1`. This variable **Form1** is an instance of the class **TForm1**.

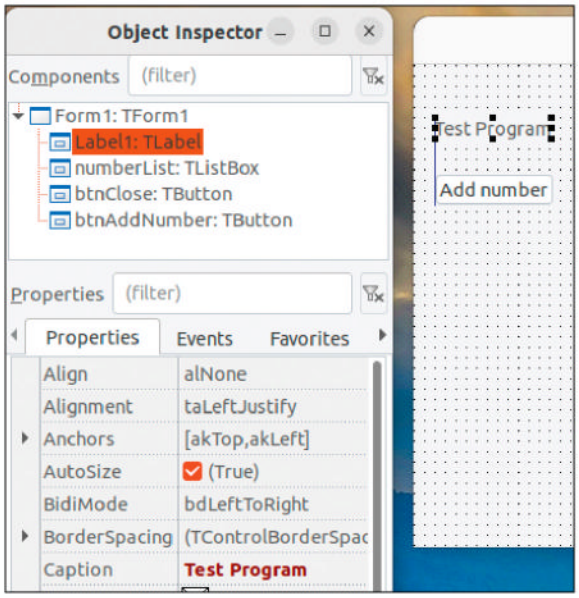
When you view the **Unit1.pas** file, you'll see a class definition like below. Each control is added here when you drop it on the form. If the names are the originals, like **button1**, find the **Name** property in the Form View and rename them to more meaningful names.

```
TForm1 = class(TForm)
  btnClose: TButton;
  btnAddNumber: TButton;
  Label1: TLabel;
  numberList: TListBox;
 private

 public

 end;

 var
  Form1: TForm1;
```

Press F12 and go back to the Form View. Double-click on the Close button. You should see it switch to the Code View and generate this code in the PAS file:

```
procedure TForm1.btnCloseClick(Sender: TObject);
begin

end;
```

Type the word `close;` between the `begin` and `end` and do Ctrl+s to save it. When you run it and click the Close button, it closes both the Form and the Application. In the Object Inspector look at the **OnClick Event** for the Close button (on the Events tab) and you'll see it has **btnCloseClick**, which is the procedure name. Event properties glue controls to code. There are events for handling mouse clicks and key presses, resizing controls and dragging and dropping.



This is the Lazarus Object Inspector. It displays properties and events for whichever control is selected.

A procedure is the same as a void function in C/C++. Here it's a method of the **TForm1** because of the **TForm1.** prefix. You'll also see that the **TForm** definition now has a copy of the method definition added to it. It's the way Pascal works with methods. There's a definition in the class definition and then the body of the method later on.

What does this program do?

When you click the Add Numbers button, it generates 10,000 random numbers and displays them in the list box. It's not quite finished yet.

Rename the button text to Add Numbers then double-click it to generate the empty method. This should add the definition in the class:

```
  procedure btnAddNumberClick(Sender: TObject);
```

In the body in the code, type `ShowMessage('Test');`:

```
procedure TForm1.btnAddNumberClick(Sender:
TObject);
begin
 ShowMessage('Test');
end;
```

The **ShowMessage('Test')** procedure call is a system procedure from the **Dialogs** unit that displays a pop-up window with the text. It's a simple way to debug but we'll also see how to use the debugger later.

Now compile the program with Ctrl+F9. If it compiles OK, you'll see the message window turn green – or red if the compile fails.

You can skip this compile stage and just press F9, which compiles and runs it, so long as the compile succeeds. If you like clicking icons, there's a green triangle just under the Save icon that also runs the program. You should see the form open and when you click the Add Numbers button, a pop-up window saying Test appears.

That's not adding numbers yet, so now remove the `ShowMessage()` and replace it with:

```
AddNumbers(10000);
```

There isn't a procedure **AddNumbers** yet, so we also have to add that.

Type in the following on the line after the `{$R *.lfm}`:

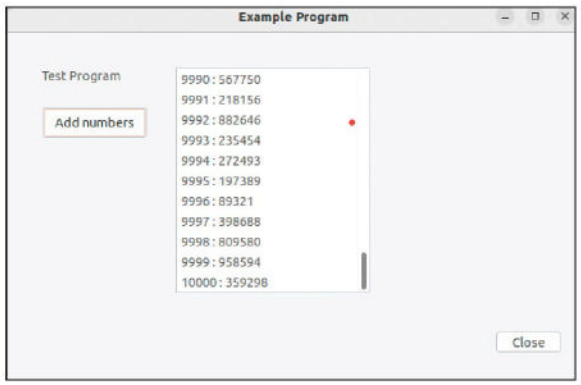
```
Procedure TForm1.AddNumbers(howMany : integer);
Begin
//
End;
```

This is going to be a method because it needs to access the **numberList** list box. Now you could type the first line in the **TForm** definition, but the IDE does that for you. Right-click on the Source Editor and when the pop-up menu appears, move the cursor over Source on the menu, then on the submenu that appears, click Complete Code.

The `//` is a single-line comment, just like in C/C++. If you want a block comment over several lines, you can use `{ ... }` or `(* ... *)`.

What about AddNumbers?

This procedure adds the specified number (passed in as **HowMany**) of random numbers into the list box. It also includes a small function **RandomNumber** that generates a number between one and a million. The



The running program after the Add Numbers button was clicked and scrolled to the bottom of the list.

Random(n) function that it calls accesses the random number generator, generating a number between 0 and n-1, where n is the parameter – in this case, 1000000. Assigning the value to **result** is the same as **return** in a C/C++ function.

AddNumbers declares an integer called **I** to use in the **for** loop. The **for** syntax is different from C/C++ but it's the equivalent of `for (i=1;I <= HowMany;i++)`. The **numberList.BeginUpdateBounds** and **EndUpdateBounds** disable and then re-enable updating the list box. It's less important on Ubuntu but on some other platforms, not using these would make it slower because it updates the list box after each number is updated.

The **IntToStr()** function converts an integer into a string. List boxes can only display strings.

```
function RandomNumber : integer;
begin
  result := Random(1000000)+1;
end;

procedure TForm1.AddNumbers(howMany:integer);
var
  i : integer;
begin
  numberList.BeginUpdateBounds;
  for i:= 1 to howMany do
    begin
     numberList.Items.Add(IntToStr(i)+' : '+
IntToStr(RandomNumber));
    end;
  numberList.EndUpdateBounds;
end;
```

When the program is run and the Add Numbers button clicked, you should get something like the screenshot *(above)* near enough instantly. Scroll the list box to see all 10,000 numbers it generated.

Debugging the program

The easiest way to debug is to click on a line in the source editor just to the left of the line number. You should see a red circle with a question mark in it. Click the circle when you want to remove it. When you hit F9, it runs until the breakpoint, then stops there.

When it's stopped, you can view various debug windows. Click the top View menu, then near the bottom of the pop-up menu, move the mouse cursor over Debug Windows and a submenu appears.

You can step through your code by pressing F8 to step forward one line at a time. If it's on a function call, press F7 to step into it. Put the cursor on a line and F4 runs it to that line and then breaks there. There's a couple of other debug commands on the Run menu.

Bring your code to life

Lazarus and Free Pascal provide a powerful development system and are easy to get started with. There are many visual components provided and you can always write your own. Just like C++, you can write procedural or object-orientated code with Pascal and the compiler generates fast-executing code. The only criticism is that executables can be a little on the large side, but there are tools to reduce it. The example executable is 26MB.

The source code for this tutorial is zipped up in the file **Lazarus-example.zip** at **https://github.com/David-H-Bolton/Projects**. Just open the link in *Firefox,* click on the file and then on the three dots on the right, and download it. Double-click on the file in your downloads and extract the contents into a folder. Then just open **example.lpr** in *Lazarus.* LXF

» WHAT IS PASCAL?

Pascal is a programming language from the early '70s. *Lazarus*/ Free Pascal is modelled on the commercial *Delphi IDE* and includes an editor and compiler, and uses the *GDB* debugger.

How does Pascal compare to C/C++?

Think of it as a simpler C++ but with a different syntax and visual elements. The main differences in syntax are:

1. It uses `begin` and `end` instead of `{` and `}` – `{` is used for multi-line comments.
2. It uses `:=` to assign values, so `a := 10;` instead of `a = 10;`. The `:=` is pronounced 'becomes equal to'.
3. It uses a single `=` instead of `==`, and `<>` instead of `!=` in expressions. Also, you don't need brackets around expressions unless they have multiple parts. `If` is always followed by `then` after the expression. Here are some examples:

```
If a = 6 then is the equivalent of if (a==6)
If b <> 9 then is the equivalent of if (b != 9)
If (a >5) and (b <= 9) then is the equivalent of if (a>5 && b <= 9)
```

4. It uses semicolons `;` to separate statements but never before an `else`:

```
If a =6 then
begin
  DoSomething();
end
else
  WriteLn(a has an 'Incorrect value');
```

5. Pascal is not case-sensitive. So `A := 6;` and `a := 6;` are the same.
6. Variable declarations are backwards compared to C/C++ and need a var. For example:

```
var
  A : integer; // integer is same as C's int
 Str : string;
```

QUICK TIP

By default, when you run a GUI app, the main form shrinks. To make it open at the same size as at design time, click on the form, then in the Object Inspector, set the AutoSize property to false.

NEXT MONTH

LXF309 will be on sale Tuesday 14th November 2023

CREDIT: Magictorch

CONTAINERS MADE EASY!

Orchestrate your containers like a pro – fire up Podman on your home server or Pi, and blast off!

My PC is an Amiga

Get the perfect retro Amiga setup with the best emulation software, some hardware extensions and Les Pounder.

Pi isn't the only fruit

We pitch a bunch of Pi 5 alternatives against each other to see whether anything can replace the Pi as our favourite SBPC.

My other arm is robotic

We take a low-cost toy robot arm and power it up with a bit of Pi control and some DIY automation software.

Printing without the fuss

Open source printing has a bad name – we don't know why – so we look at software to make printing fuss-free.

Content of future issues subject to change. We might have been sandboxed ourselves...

LINUX FORMAT
The #1 open source mag

Future Publishing Limited,
Quay House, The Ambury, Bath, BA1 1UA
Email contact@linuxformat.com

EDITORIAL
Editor-in-chief Neil Mohr
Art editor Fraser McDermott
Production editor Katharine Davies
Group editor-in-chief Graham Barlow
Group art director Warren Brown
Editorial contributors Mike Bedford, David Bolton, Neil Bothwick, Fraser Brown, Stuart Burns, Nate Drake, Darien Graham-Smith, Matthew Holder, Dave James, Ruby P Jane, Jon Masters, Nick Peers, Avram Piltch, Les Pounder, Michael Reed, David Rutland, Mayank Sharma, Chris Szewczyk, Mike Williams
Cover illustration Magictorch.com
Raspberry Pi is a trademark of the Raspberry Pi Foundation. Tux credit: Larry Ewing (lewing@isc.tamu.edu).
Content production Adequate Media Limited

ADVERTISING
Commercial sales director Clare Dove
clare.dove@futurenet.com
Advertising director Lara Jaggon
lara.jaggon@futurenet.com
Account director Andrew Tilbury
andrew.tilbury@futurenet.com

INTERNATIONAL LICENSING
Head of print licensing Rachel Shaw
Linux Format is available for licensing and syndication. To find our more contact us at licensing@futurenet.com or view our content at www.futurecontenthub.com

NEW SUBSCRIPTIONS & BACK ISSUES
Web www.magazinesdirect.com
UK 0330 333 1113 **World** +44 (0) 330 333 1113

EXISTING SUBSCRIPTIONS
Web www.mymagazine.co.uk
UK 0330 333 4333 **World** +44 (0) 330 333 4333
Subscription delays: Disruption remains within UK and international delivery networks. Please allow up to seven days before contacting us about a late delivery to **help@magazinesdirect.com**

CIRCULATION
Newstrade & retail category director Ben Oakden

PRODUCTION AND DISTRIBUTION
Group head of production Mark Constance
Production manager Nola Cokely
Senior ad production manager Jo Crosby
Digital editions manager Jason Hudson

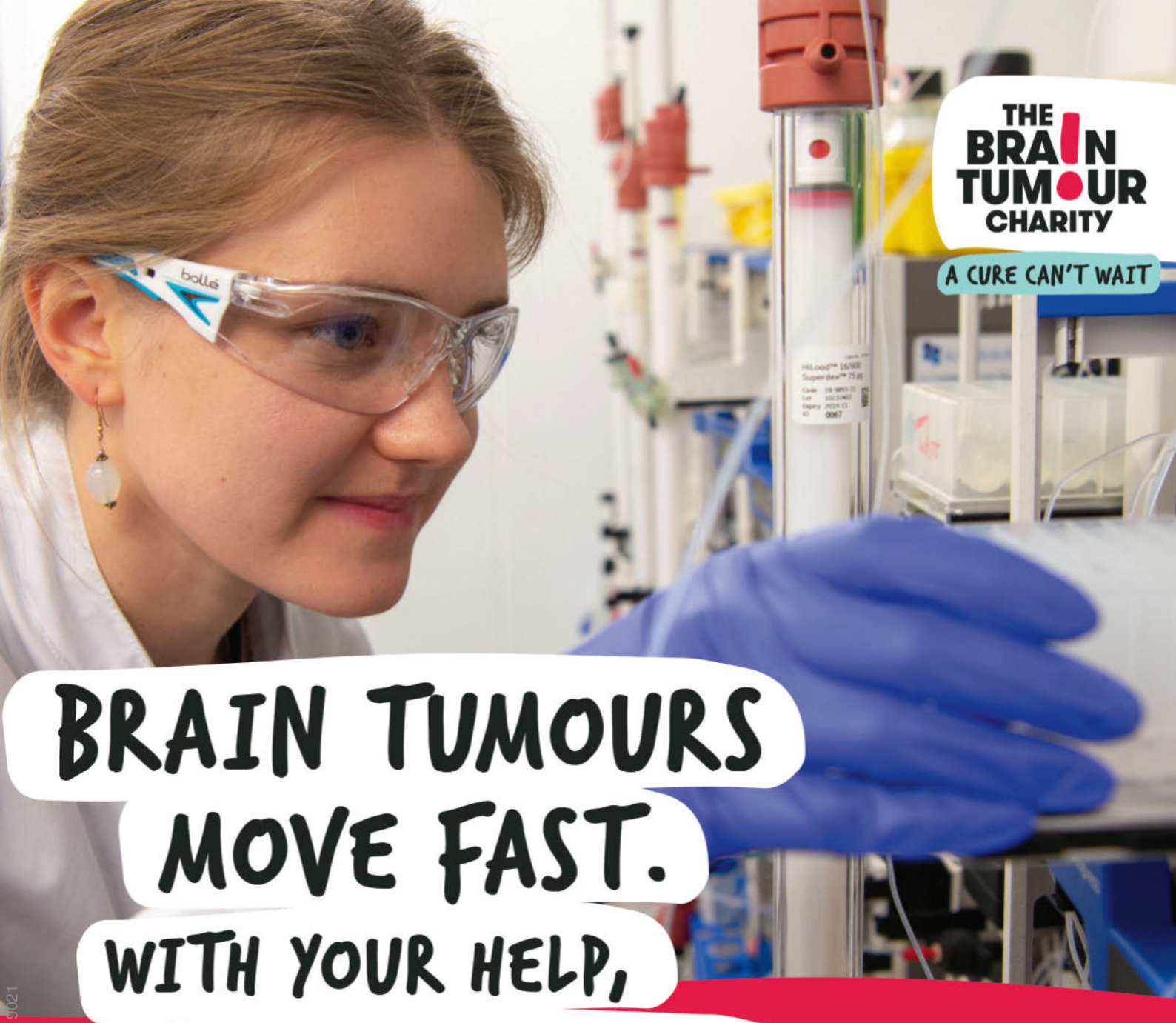
THE MANAGEMENT
Managing director technology group Paul Newman
Global head of design Rodney Dive
Commercial finance director Tania Brunning
Printed by Wyndeham Peterborough, Storey's Bar Road, Peterborough, Cambridgeshire, PE1 5YS
Distributed by Marketforce, 5 Churchill Place, Canary Wharf, London, E14 5HU
www.marketforce.co.uk For enquiries email: mfcommunications@futurenet.com

The Electronic Frontier Foundation is the leading nonprofit organization defending civil liberties in the digital world. Founded in 1990, EFF champions user privacy, free expression, and innovation through impact litigation, policy analysis, grassroots activism, and technology development. We work to ensure that rights and freedoms are enhanced and protected as our use of technology grows.

EFF.ORG

ELECTRONIC FRONTIER FOUNDATION

Protecting Rights and Promoting Freedom on the Electronic Frontier



THE BRAIN TUMOUR CHARITY

A CURE CAN'T WAIT

BRAIN TUMOURS MOVE FAST.

WITH YOUR HELP, WE CAN TOO!

We're working to create a future where brain tumours are curable.
We urgently need your help to accelerate research.

Text DEFEAT5 to 70507 to donate £5, please help us to find a cure.

thebraintumourcharity.org

9021

9000